UIUCDCS-R-74-631

OCOMO
OPTICAL CORRELATION MODULATION

by

Robert L. Budzinski

March, 1974

DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN · URBANA, ILLINOIS

UIUCDCS-R-74-631

OCOMO
OPTICAL CORRELATION MODULATION

by

Robert L. Budzinski

March, 1974

Department of Computer Science
University of Illinois
Urbana, Illinois 61801

This work was supported in part by Contract No. N000 14-67-A-0305-007 and was submitted in partial fulfillment of the requirements for the degree of Master of Science in Electrical Engineering, March, 1974.

ACKNOWLEDGMENT

The author wishes to take this opportunity to express his gratitude to his advisor, W. J. Poppelbaum, for suggesting this thesis, for his support, and his guidance and friendship.

He is very thankful for the fine work done by the fabrication group directed by Frank Serio and by the drafting department directed by Mark Goebel. Special thanks go to Evelyn Huxhold for typing this thesis.

He also wishes to thank his wife, Lyn, for all her assistance and patience.

TABLE OF CONTENTS

CHAPTER 1

GENERAL PRINCIPLES OF OCOMO

1.0 Introduction

Optical COrrelation MOdulation is a scheme used to implement the encoder/decoder section of a digital voice communication system. A one-to-one code is used so that the codewords can be optically correlated to a set of codewords, in order to obtain the original voice-signal amplitude. Figure 1.1 shows a block diagram of the system: The system converts the sampled analog signal to binary sequences with a length of 6 bits. The optical encoder maps a 6-bit sequence into a codeword with a length of 8 bits. This is done on-line. The 8-bit long codewords are then transmitted serially. Upon reception, the optical correlator compares the received codeword with (parts of) the whole set of codewords: The correlator thus remaps the codeword on-line back into the original 6-bit sequence. This sequence is then converted to the original amplitude by means of a D/A converter and subsequent filtering. The bandwidth of OCOMO is roughly 7.5 KHz for the voice input signal.

1.1 Pulse Code Modulation

Pulse code modulation is a digital modulation scheme in which an (analog) signal is represented, a sample at a time, by a group of digital pulses. The signal which is to be coded is sampled at a rate which is at least twice the maximum frequency of the signal (Nyquist's Theorem). This sampled signal then has its amplitude quantized: In OCOMO, 64 levels of quantization are used. Associated with each quantized level is a codeword or a train of pulses, with N pulses comprising a codeword. Since 2^6 or 64 levels are used in OCOMO, at least 6 bits are needed for each level. In short: PCM takes an analog signal (with a bandwidth restriction!), samples it at a fixed rate, quantizes the

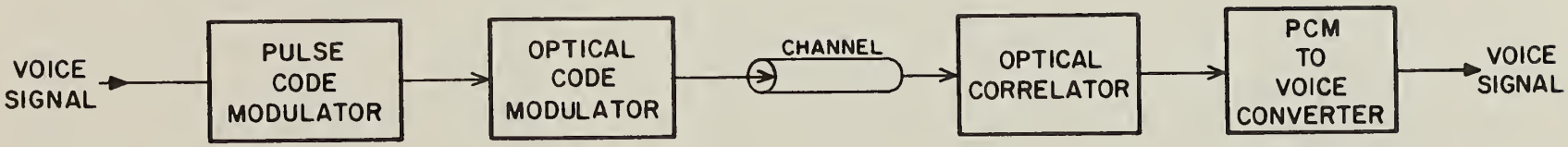

Figure 1.1 OCOMO Block Diagram

amplitude of the sampled signal, and then transmits a codeword associated with the quantized amplitude of the sampled signal.

Let us examine why the signal must be sampled at a rate equal to twice the maximum frequency of the input voice signal, i.e. why Nyquist's Theorem must be applied. Nyquist's Theorem states:

> All the information in a signal bandlimited to a frequency f_{max} is contained in a sample of the signal taken every $\frac{1}{2f_{max}}$ seconds.

In a qualitative sense the theorem seems reasonable since if one samples too slowly, the signal may change "behind our back." A more mathematical justification is readily available:

Let $g(t)$ be a signal to be sampled and let $G(\omega)$ be $g(t)$'s Fourier transform. $G(\omega)$ is pictured in Figure 1.2a and its bandwidth is f_{max}.

Figure 1.2b shows $G(\omega)$ as a periodic function. Then

$$G(\omega) = \frac{1}{4\pi f_{max}} \sum_{n=-\infty}^{\infty} C_n \, e^{(jh\omega/2f_{max})} \qquad |\omega| \leq 2 f_{max} \tag{1-1}$$

$$\text{with } C_n = \int_{-2\pi f_{max}}^{+2\pi f_{max}} G(\omega) \, e^{(-jn\omega/2f_{max})} \, d\omega \tag{1-2}$$

$$\text{but } \quad g(t) = \frac{1}{2\pi} \int_{-2\pi f_{max}}^{+2\pi f_{max}} G(\omega) \, e^{j\omega t} \, d\omega \tag{1-3}$$

$$\text{in particular } \quad g\left(\frac{-n}{2f_{max}}\right) = \frac{1}{2\pi} \int_{-2\pi f_{max}}^{+2\pi f_{max}} G(\omega) \, e^{(-jn\omega/2f_{max})} \, d\omega = \frac{C_n}{2\pi} \tag{1-4}$$

Thus, if one knows $g(t)$ at times, one can reconstruct the Fourier series of $G(\omega)$, and if one knows $G(\omega)$, one can reconstruct $g(t)$. If one applies the above processes, one finds that

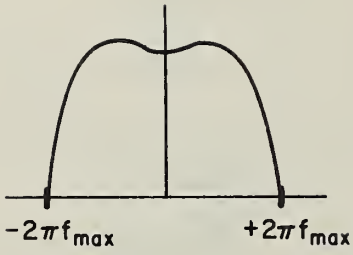

Figure 1.2a Signal Spectrum

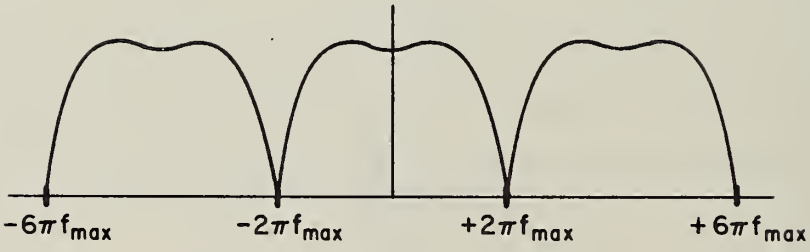

Figure 1.2b Periodic Signal Spectrum

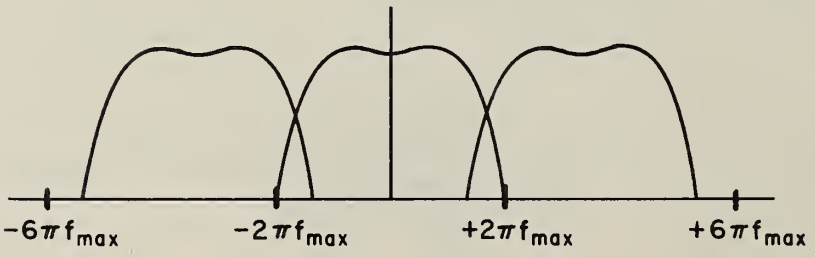

Figure 1.2c Periodic Signal Spectrum with Aliasing

Figure 1.2 Spectra

$$g(t) = \sum_{n=-\infty}^{\infty} g\left(\frac{n}{2f_{max}}\right) \frac{\sin(2\pi f_{max}(t - n/2f_{max}))}{2\pi f_{max}(t - n/2f_{max})} \qquad (1\text{-}5)$$

This reconstruction of g(t) thus corresponds to passing the samples of g(t) through an ideal low-pass filter with bandwidth f_{max}!

Figure 1.2c shows a sampling rate of less than $\frac{1}{2f_{max}}$. The resulting aliasing or overlapping of the frequency spectrum waveforms will result in loss of information and therefore distortion. When sampling is done at less than the Nyquist rate, the C_n's of equation (1-4) are no longer equal to a single discrete sample, but rather they are functions of the previous samples and the next sample. Since proper reconstruction would require an anticipating filter (which is physically unrealizable), some of the information must be lost.

Inherent to PCM is quantization noise, caused by quantizing the sampled signal. The RMS value of this noise is:

$$N = \left(\frac{1}{a}\right) \int_{-a/2}^{a/2} M^2 dM = \frac{a}{2\sqrt{3}} \qquad (1\text{-}6)$$

where a is the difference in voltage between two quantization levels. The peak signal is Ma, M being the number of levels. The voltage signal to noise ratio is $2\sqrt{3}$ M. For OCOMO, M is 64: The signal to noise ratio is found to be 221.7.

1.2 The OCOMO Encoder

Basic to all pulse code modulators is a circuit to sample and hold the desired signal, a circuit to quantize this sample, and a circuit to digitize this quantized sample. One way to accomplish the above process is to use a Tracking A/D Converter. This is a converter which varies the digital representation to match the analog signal. To ensure good tracking, one must be sure that the slew rate of the A/D Converter is not exceeded by the slew rate of the signal. The maximum slew rate of the signal is given by:

$$\frac{d}{dt}\left[V_{max} \sin(2\pi f_{max} t)\right] \qquad t = \text{a multiple of } \frac{1}{f_{max}} \qquad (1\text{-}7)$$

In order to sample, quantize, and digitize, one only has to store the digital signal from the A/D Converter. This is the method used in OCOMO.

After digitizing the sample, one could just transmit the digital representation, or one could encode it. One can encode for basically two reasons: One reason would be to add error-correction capability. The other reason would be the desire to encode the data in order to ensure secrecy. In OCOMO the latter reason predominates.

One straight-forward way to encode is to use the digitized amplitude to address a codeword from a memory. If one uses a semiconductor random-access memory (RAM), the memory must always be energized, or else data must be replaced whenever the system is repowered. This problem can be avoided by using a read only memory (ROM). But an electrical ROM is generally difficult to change. A good compromise between a RAM and an electrical ROM is an optical ROM of the type used in OCOMO. This memory is programmed by placing a plastic transparency with the codeword bits in the form of opaque and transparent rectangles on a filmstrip. An example is shown in Figure 1.3. In this figure, each column is a codeword, i.e. each column is associated with a digitized amplitude. In what follows, these columns will be called codestrips.

The memory works as follows. The (digitized) quantized signal is reconverted to an analog form by using a D/A converter, and this signal is used to horizontally deflect the beam of an X-Y oscilloscope. The beam is vertically deflected using a staircase generator. Visibly, when a signal is sampled, the resulting quantized voltage selects a codestrip. Then the beam is deflected so as to scan each bit position of the codeword. To reconvert to an electrical signal, a photo-detector is used.

1.3 OCOMO Decoder

The basic procedure used for decoding, is to compare the received codeword with the complement of all the codewords optically. The following

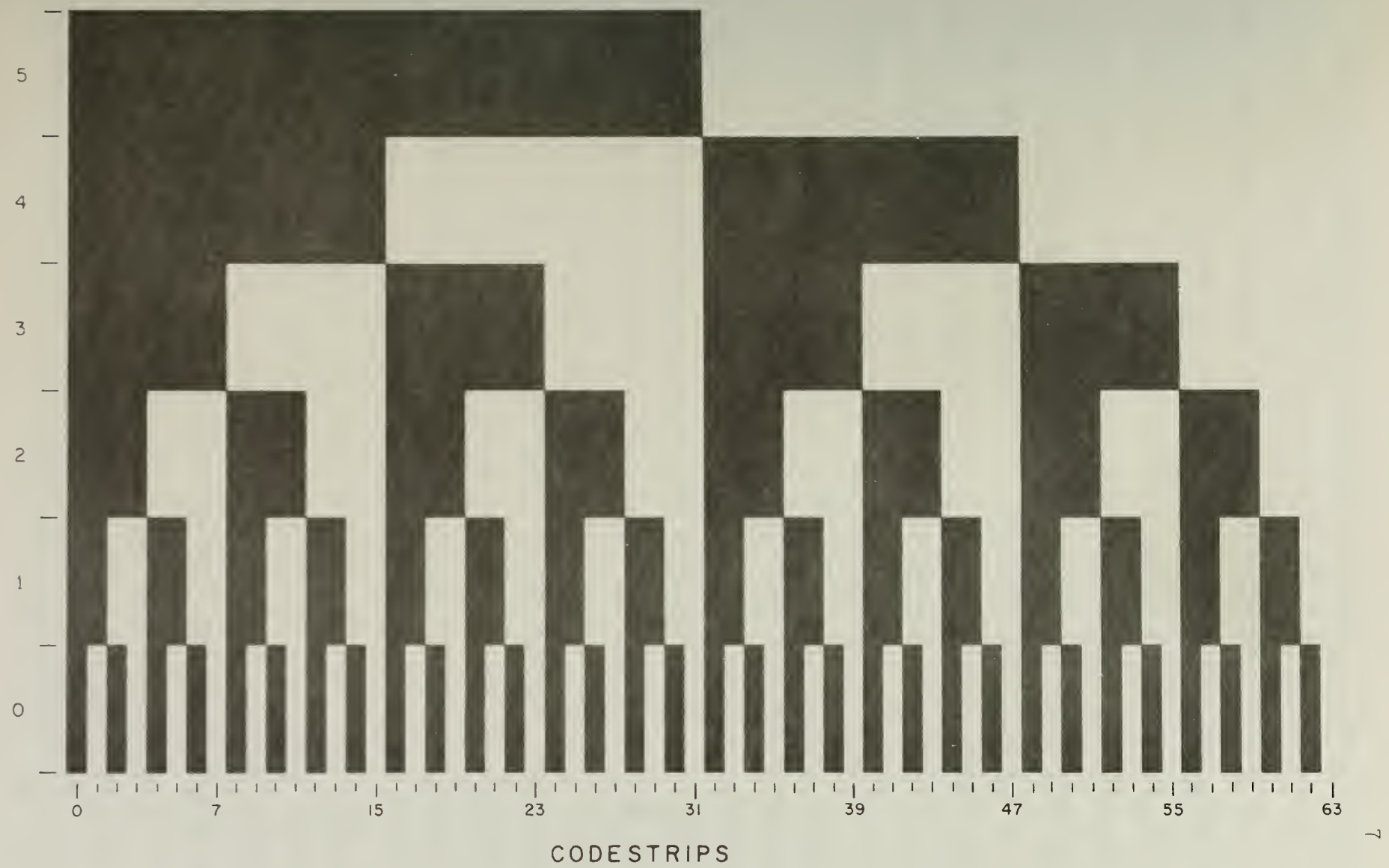

Figure 1.3 OCOMO Code Mask

convention is made: A logical "0" corresponds to a transparent spot on the codestrip, and a "1" corresponds to an opaque spot. If a particular bit position in the received codeword is a "0", the corresponding bit position on the complementary codestrip of the received codeword will be dark. If a bit position is a "1", the corresponding position will be transparent. If the first bit position of the received codeword is a "0" and the first bit position of the complement of all the codestrips is optically scanned, some of the codestrips, including the one associated with the codeword, will not pass light. If every bit position of the received codeword which has a "0" is similarly scanned, the strip corresponding to the codeword will not pass any light. In order for the above correlation scheme to unambiguously decode, every codeword must have its "0"'s placed so that when the procedure is performed, only the codestrip associated with the codeword does not pass light.

Using the above decoding scheme puts certain restrictions on the code. The "all-0-bit" codeword cannot be used because its complementary codestrip would be opaque, and therefore would never pass any light! Thus it would always be decoded as a received word! If we want 64 codewords, we would need at least 7 bits. If one of the 7 bit codewords had one "0" and six "1"'s, when decoding, only one row would be scanned: To differentiate this word from the other 63, a transparent spot would be required in the corresponding bit of the other 63 words. Thus,these other words would have only 6 independent bits. Now in order to get 63 words from 6 bits, at least one of the words would have only one "0" and five "1"'s. But then this would allow at most 5 independent bits and we could not find enough words. In other words: <u>All words with a single "0" must be excluded</u>. Similar reasoning shows that words with two "0"'s and five "1"'s must also be excluded. If we use 7 bit codewords with three "0"'s, we have $\binom{7}{3} = 35$ decodable codewords. If one tries to include codewords with four "0"'s, a decoding ambiguity will arise. There are $\binom{7}{4} = 35$ codewords with four "0"'s.

For every four-"0"-codeword in the set, there are $\binom{4}{3}$ = four three-"0"-codewords whose "0" positions match the "0" positions of the four-"0"-word. If such a four-"0"-word is received, the decoding scheme will determine that the four-"0"-word and any of the matching three-"0"-words in the set is the word received. If any four-"0" word is entered into the set of codewords, four of the three-"0"-words must be eliminated. If a second four-"0"-word is added, at least three more three-"0"-words are eliminated. Every other four-"0"-word added, eliminates at least one new three-"0"-word, until all the three-"0"-words have been eliminated. Adding three-"0"-words and eliminating four-"0"-words leads to the same result. Thus, fewer codewords result from combining the three and four "0" words than not combining the groups!

It has thus been shown that the all-"0"'s-codeword cannot be used, as well as the codewords with one "0" or two "0"'s. Also, it has been proved that the three-"0"-codewords and four-"0"-codewords cannot be used together. This eliminates the following number of codewords: $\binom{7}{0} + \binom{7}{1} + \binom{7}{2} + \binom{7}{3} = 64$. This leaves us with at most 64 usable codewords. But the all-"1"-codeword cannot be used either, because this word would cause all of the bit positions to be scanned. (This word could be used if it were assigned to a constant level or special hardware were added to change its level, but this would defeat the ease of changing codeword to level assignments which is the main advantage in using an optical ROM). Therefore, codewords 8 bits long are used.

If 8 bits are used, a set of codewords with four "0"'s can be used with the OCOMO decoding scheme. First, $\binom{8}{4}$ = 70 codewords are available. The scheme works, because checking the four "0"'s of the received word will differentiate the complementary strip associated with the received word from the other complementary strips. This can be shown as follows. Since the complement of "0" is an opaque spot, the strip associated with the received word will pass no light. Every other strip must pass light because some of the four "0"'s or

opaque spots of every other codeword strip must be in different positions: For if all the positions matched, it would have to be an identical strip.

The exact decoding procedure used in OCOMO is as follows. There are 64 flipflops associated in a one-to-one correspondence with the 64 codestrips. After receiving a codeword, each row corresponding to a "0" bit position is scanned. Before any strips are scanned, all 64 flipflops are initialized to the "1"-state. As each strip is scanned, the appropriate flipflop is enabled if light passes from the particular strip. The circuitry is designed so that if a particular strip passes light more than once, the appropriate flipflop changes state only once. After all 4 x 64 positions are checked, one flip-flop will be in the initialized state and the other 63 will have changed state. Thus, the received codeword is associated with the flipflop which has not changed. Using a 1 of 64 to 6 encoder, a buffer, a D/A converter and a filter, we complete the decoding.

A refinement of the above decoding procedure was conceived after OCOMO had been built. This decoding scheme places no restrictions on the code, so that 6 bits would be sufficient for 64 codewords. The scheme is as follows: After receiving the 6-bit codeword, the decoder checks the codestrips, codeword by codeword. A particular codeword is checked until there is a discrepancy between the received word and the codestrip: If each codestrip is arranged in the order that it was at the encoder, one merely has to keep track of the number of codestrips that were checked prior to finding the matching codestrip!

This second scheme actually requires less checking of codestrip bit positions than the first. The first method required 4 x 64 = 256 checks. The second method requires only 126 checks in the worst case; this worst case occuring when the received codeword corresponds to 64. Thus all codestrips must be checked at least once. More precisely: The first bit of the 64 code-words must be checked; of these 32 will agree with the received codeword.

Therefore, the second bit must be checked for 32 of the codestrips. Similarily the third bit of 16 strips must be checked, etc. So, in the worst case

$$64 + 32 + 16 + 8 + 4 + 2 = 126$$

positions must be checked.

1.4 Pulse Code Modulation as a Function of Time

There is a straight-forward way to make the quantized signal which selects the codeword to be transmitted as an implicit function of time: One can keep a running MODULO-64-sum of the PCM codewords, and this (newly updated) sum can select the codeword to be transmitted! At the decoder, the PCM level decoded will actually represent the sum: If the previous sum is known, the latest PCM level can be found by subtracting the old sum from the new sum. In this way the one-to-one correspondence between the PCM signal and the OCOMO codewords varies, depending on what the input voice signal to be coded is. Since this signal is a function of time, the mapping of the PCM codeword to the OCOMO codeword is an implicit function of time. This scheme was not implemented.

CHAPTER 2

OCOMO ENCODER, A CIRCUIT DESCRIPTION

2.0 Introduction

The Block diagram of the OCOMO Encoder is shown in Figure 2.1. The audio amplifier has an Automatic Gain Control (AGC) to ensure that the quantization range of the A/D Converter is not exceeded. The output of the Tracking A/D Converter is sampled and stored in the digital Sample and Hold, which serves as a buffer register. (The output of the D/A is the quantized, sampled representation of the input signal!) When applied to the horizontal section of the CRT, this voltage selects one of 64 codestrips placed on the CRT. The synchronized staircase generator then deflects the beam past the bits of the code strip. The light output is detected by a photomultiplier tube, and its (digital) electric output is transmitted. Figure 2.2 shows a set of 64 codestrips as they are arranged on the face of the CRT.

2.1 Audio Amplifier

Figure 2.3 shows the audio amplifier circuit. The input signal enters the circuit through a unity-gain buffer. Op Amp 1 is used to amplify the signal to a peak amplitude of 1 to 2 volts. The AGC circuit controls the gain by varying the operating point of the four diodes at the input of Op Amp 2. This Op Amp is used to reamplify the signal, and the differential inputs are connected so that the voltage setting the operating point of the diodes produces a common mode. The circuitry after the stage described above is used to rectify and integrate the signal. The output of the Integrator sets the operating point of the diodes. The transistor is used for current gain. The 10K-pot in the Integrator amplifier determines the no-signal operating point of the diodes. Op Amp 3 is used to set the peak-to-peak amplitude of the signal to 10 volts, the quantization range of the Tracking A/D Converter.

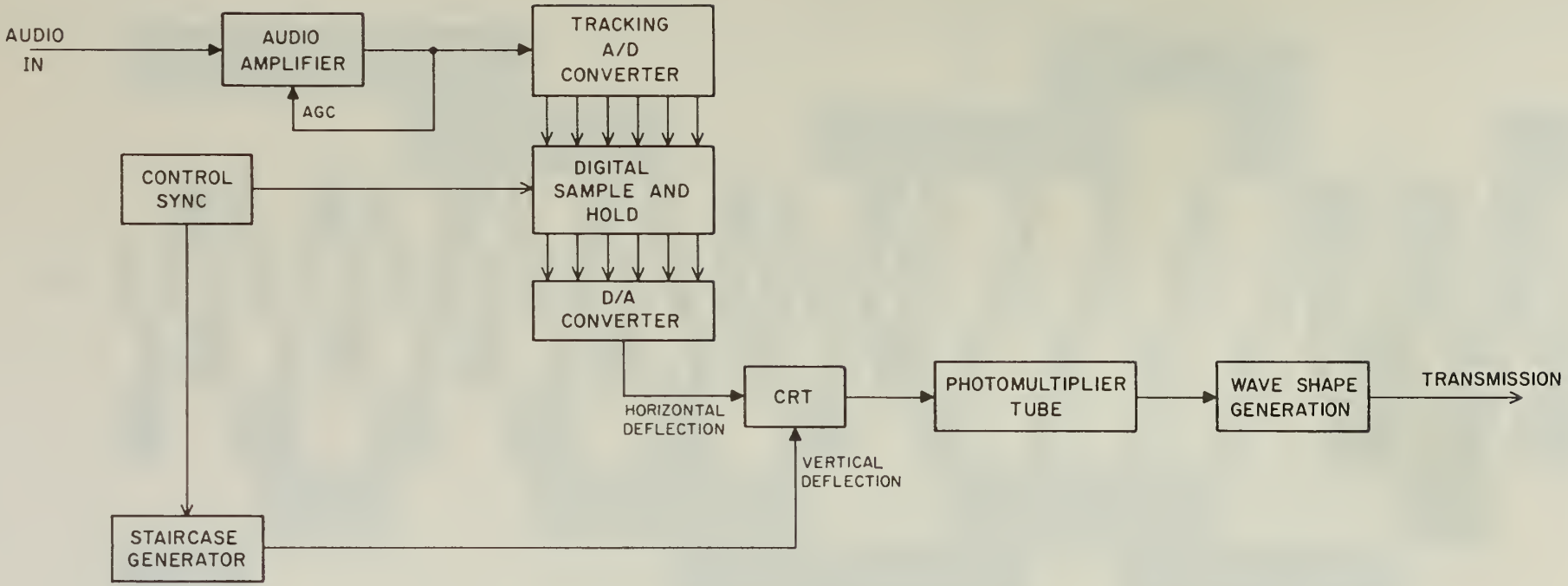

Figure 2.1 Encoder Block Diagram

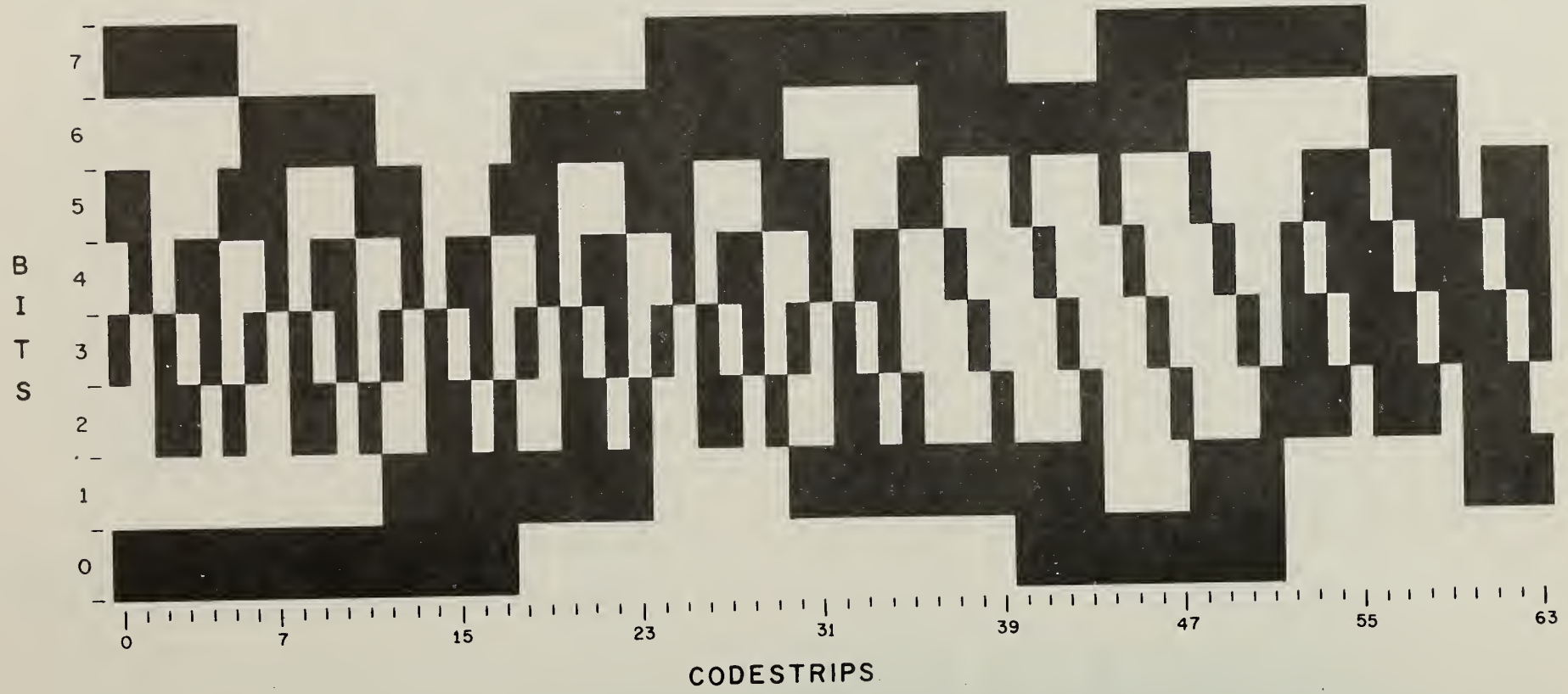

Figure 2.2 OCOMO Codestrips

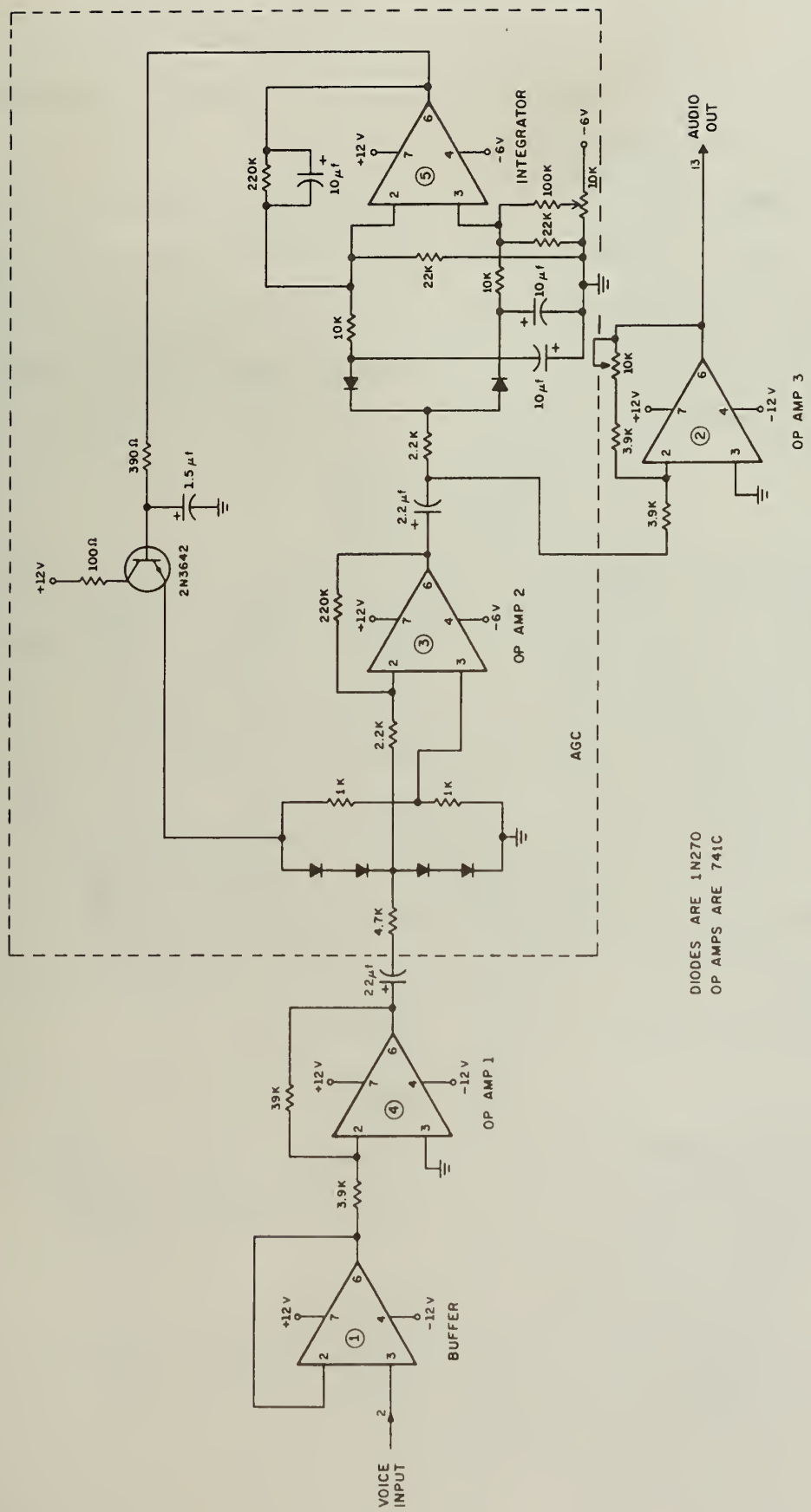

Figure 2.3 Encoder Voice Amplifier

2.2 The Horizontal Deflection Circuit

Figure 2.4 shows the circuitry for the Tracking A/D Converter, the digital sample and hold, and the D/A Converter for driving the horizontal deflection amplifier of the CRT.

The input stage of the Tracking A/D Converter is used as a DC level shift and buffer. The D/A Converter sinks current from the buffered input signal. The 72510 Voltage Comparators monitor the voltage drop from the 4.7K + variable 1K resistor. The difference in voltage between the number 3 pin of each 72510 is set to be one quantization level, which is

$$Q_{level} = \frac{V_{max}}{N_{level}}$$

The V_{max}, the maximum quantization level is 10 volts. The N_{level}, the number of levels, is 64. In this converter Q = 10/64 = .156 volt. Pin 3 of the upper comparator is set at +.078 volt. The other Pin 3 is at -.078 volt.

The A/D Converter is designed to keep the voltage at pin 2 of the comparators between $\pm$.078 volt. If the voltage is above .078v, the 74191 counters count down. This increases the current that the MC 1406 D/A sinks. If the voltage is between $\pm$.078, the counters are disabled. If the voltage is below -.078 volts, the counters count up and thereby decrease the current the D/A sinks.

The maximum slew rate of the A/D Converter is:

$$Q_{level}/T_{clock}$$

where T_{clock} is the period of the clock driving the counters. Since the clock functions at 3 MHz, T_{clock} is .33 μs. Therefore, the slew rate is

$$\frac{.156}{.33} \times 10^{-6} = .468 \text{ v/μs}$$

The maximum slew rate of the input signal is the derivative of the signal at the maximum voltage and frequency or:

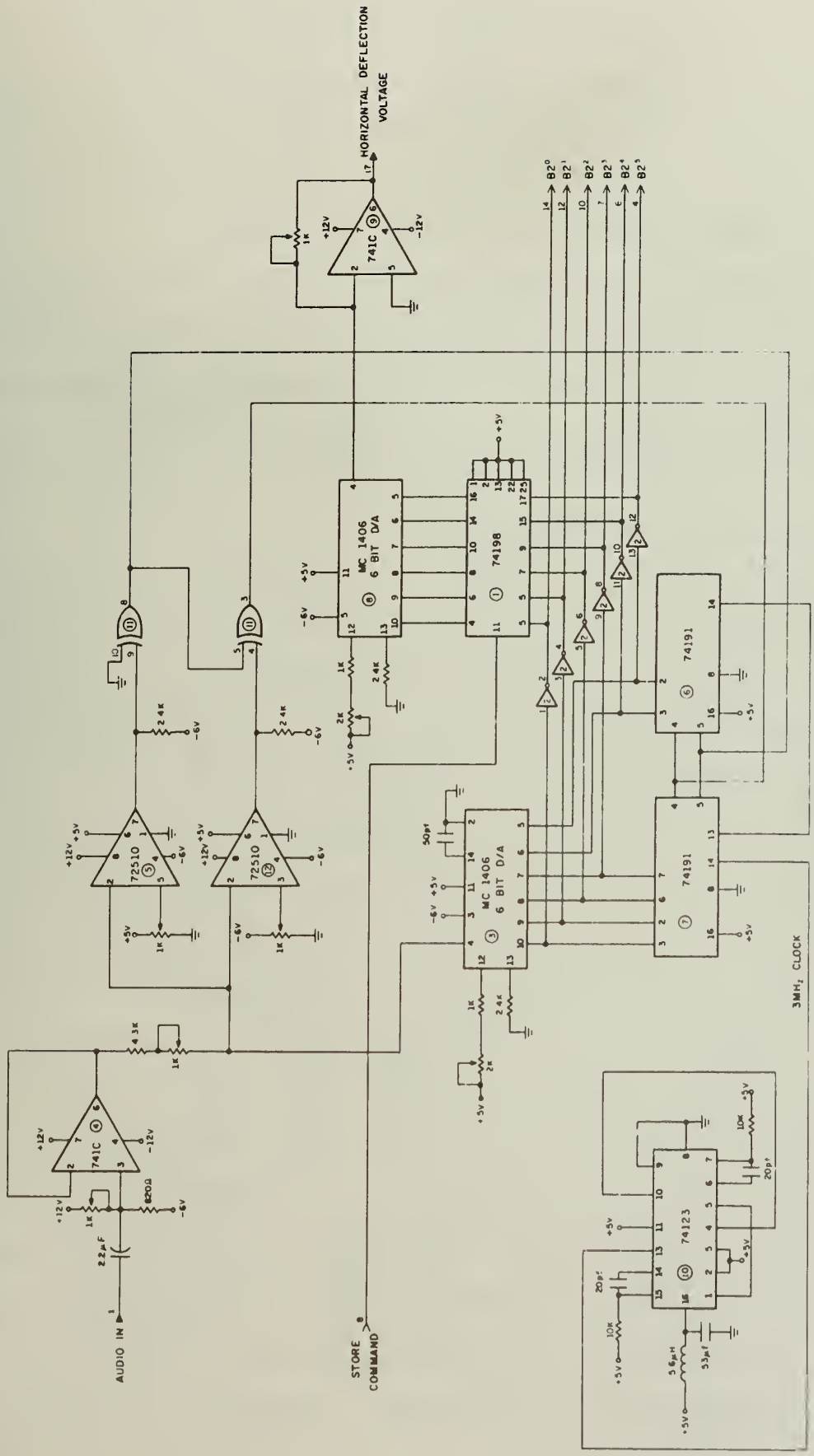

Figure 2.4 Tracking A/D Converter

$$\frac{d}{dt}\left[5 \sin (2\pi \times 7.5 \times 10^3 t)\right]$$

$$= 5(2\pi \times 7.5 \times 10^3) \text{ volts/sec}$$

$$2.36 \times 10^5 \text{ volts/sec}$$

$$= 0.236 \text{ volts}/\mu s$$

Therefore, the A/D Converter can track the input signal.

The signal is then sampled and held digitally. This avoids the cost and error introduced by an analog sample and hold circuit. The quantized sampled and stored digital representation of the input signal is then reconverted to an analog voltage in order to select a codestrip in the optical ROM.

2.3 The Vertical Deflection Circuit

Figure 2.5 contains the vertical deflection circuit, the optical ROM output processor, and the control signals for the vertical and horizontal deflection circuits. Figure 2.6 shows the vertical deflection voltage, the horizontal control signal, a possible output sequence using the combined sync signal, and the clock signal.

The vertical deflection voltage is generated by converting the digital output of a counter to an analog voltage. The main system clock drives the counter. Note that the counter converges to the desired state without regard to the initial state: When the counter reaches a maximum (minimum) of all "1"'s (all "0"'s), the Q output of the JK flipflop goes to a "1" on the next falling edge of the clock. This disables the counter for one clock pulse. It also loads in the value "1110" ("0001") into the counter. This loading changes the direction of counting. After the clock period, the Q of the flipflop goes to "0". The counter then counts down (up) to "0000" ("1111"), and the process reoccurs. Only the lower 3 order bits of the counter are used in the conversion to the analog voltage. The horizontal control pulse is generated by the NAND of Q and the inverse of the MAX/MIN output of the counter. The length of the control pulse is set by a monostable.

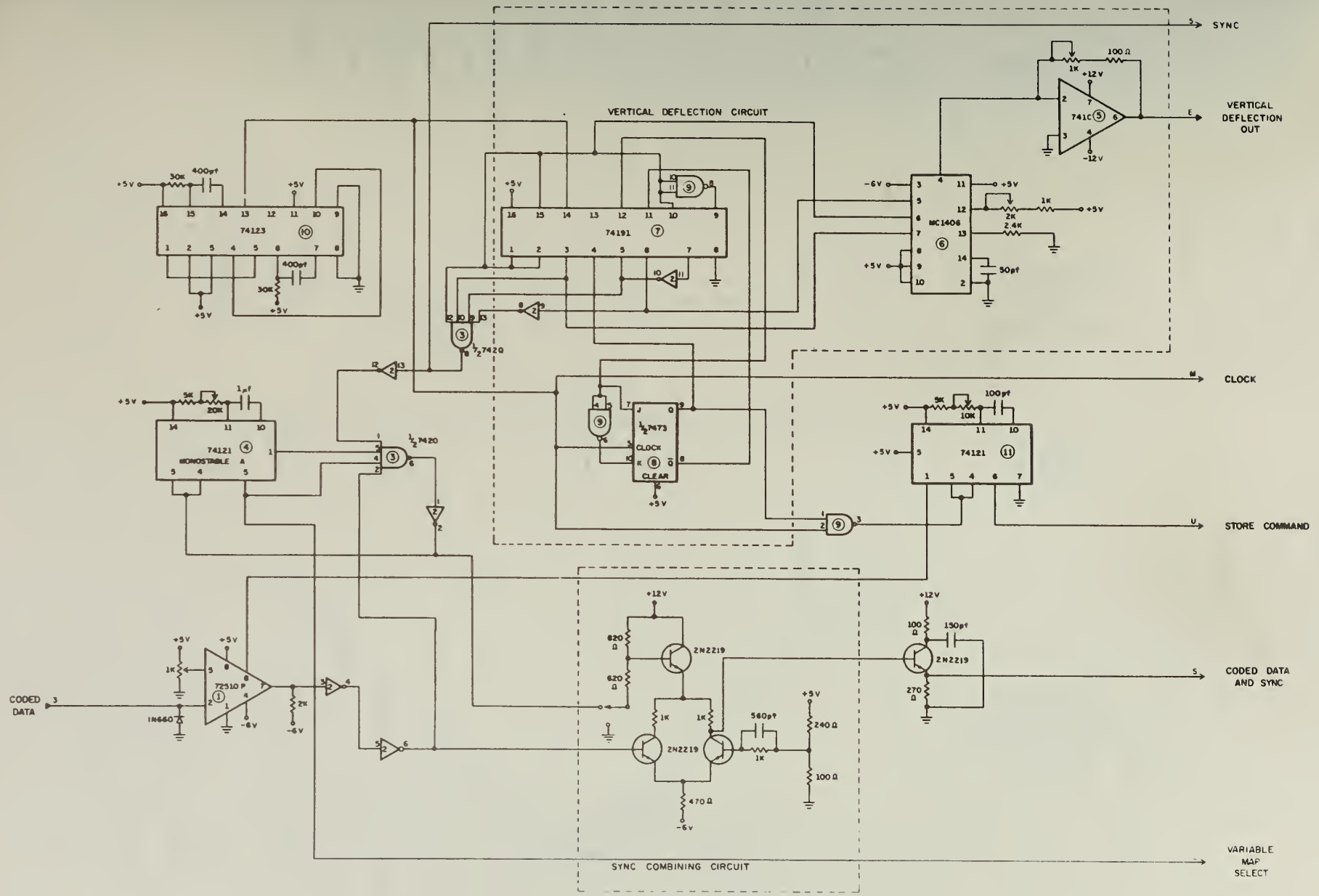

Figure 2.5 Vertical Deflection Card

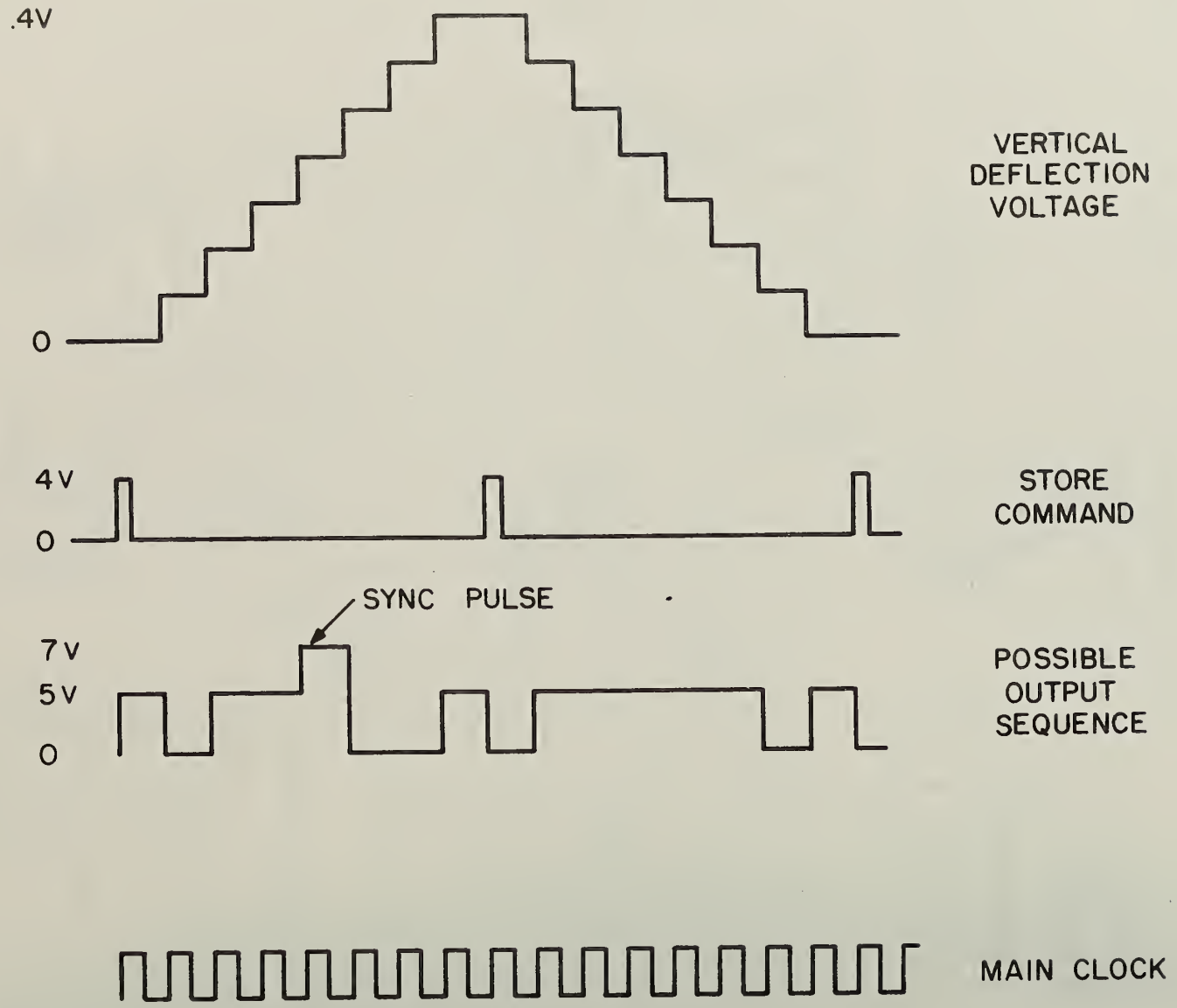

Figure 2.6 Encoder Timing

Figure 2.7 shows the circuit for the 931A photomultiplier tube. The phosphor for the encoder CRT is P24. Its persistence fall time is about 1.5 μs. Since the audio signal to be coded is band limited to 7.5 KHz, the sampling rate must occur at a frequency of at least 15 KHz or about once every 64 μs. This in 64 μs, 8 bits must be read out of the ROM or 1 bit in 8 μs. Thus, the persistence time is acceptable. The anode of the photomultiplier tube sinks current: The current is converted to a voltage using a resistor; since the data rate is relatively slow, this is acceptable. This voltage is then digitized using a comparator.

There are two modes of synchronizing the encoder to the decoder. One mode is to use a particular state of the vertical counter: The value we chose is binary 3. The other mode is to combine the sync signal with the data from the optical ROM. In this mode the sync signal is combined by raising the data voltage above the normal "1" voltage by two volts. The sync signal can then be combined only if the data is in a "1" state, so as to avoid entering a false "1" into the data to be transmitted. A third condition can be added to the transmission of a sync pulse. Namely, that monostable A be in the untriggered state. This monostable is triggered after each sync pulse is sent: When this third condition is added, the approximate frequency of sync pulses can be controlled, because there is no guarantee that the data will be in the "1" state when the other conditions for sending a sync pulse are met. The sync signal is combined with the data by changing the DC supply voltage of a differential amplifier. When the sync pulse is added, the supply voltage is raised 2 volts. The output of the differential amplifier is then connected to a transistor used as a transmission line driver.

2.4 Encoder Power Supplies

Figure 2.8 shows the -1050 volt power supply used to operate the photomultiplier tube. The supply is a basic series transistor regulator. The design output current is 10 ma. The normal operating current is approximately 5 ma. At 5 ma of current, the output tipple is .15 volt peak to peak.

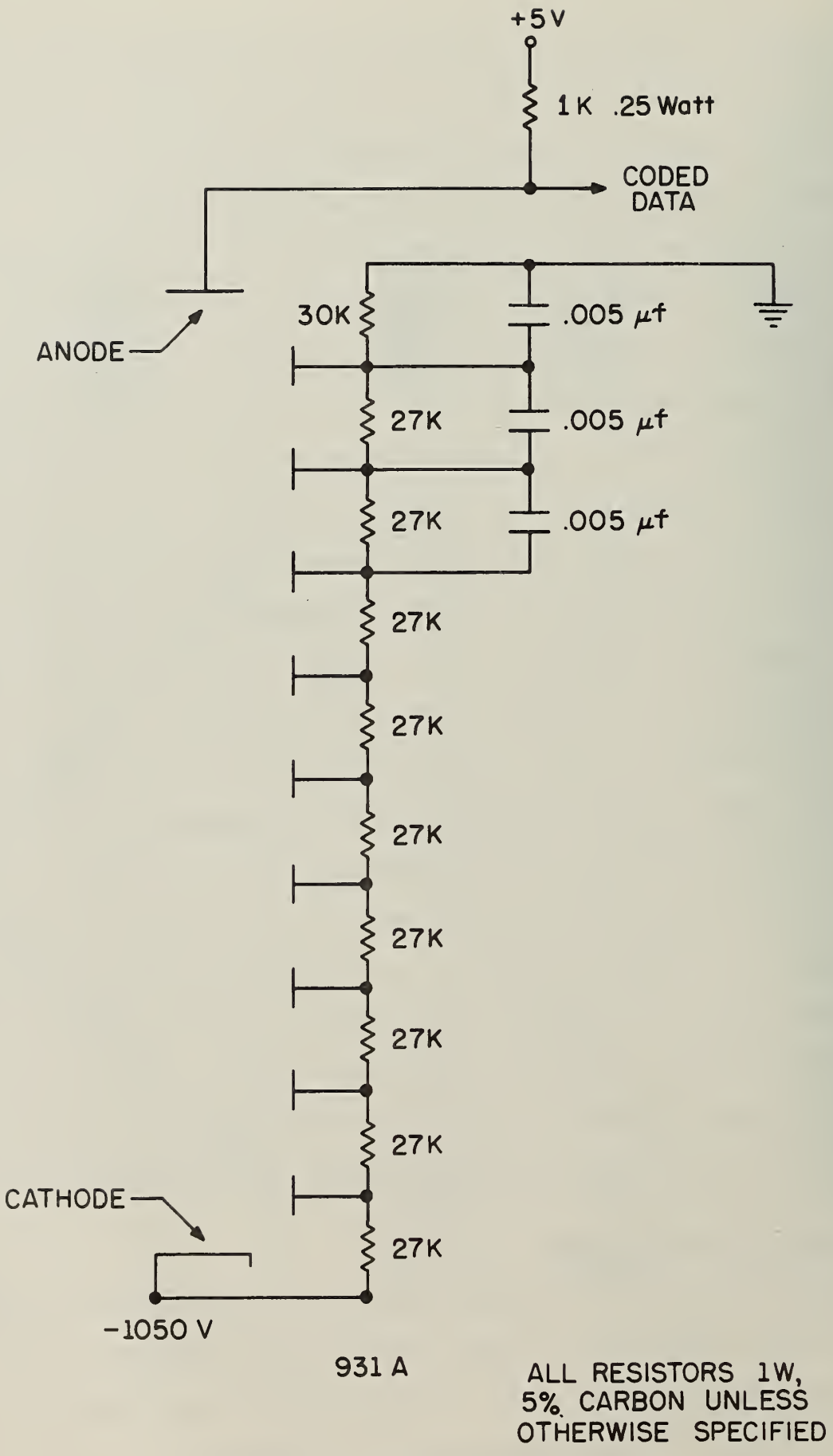

Figure 2.7 Photomultiplier Circuit

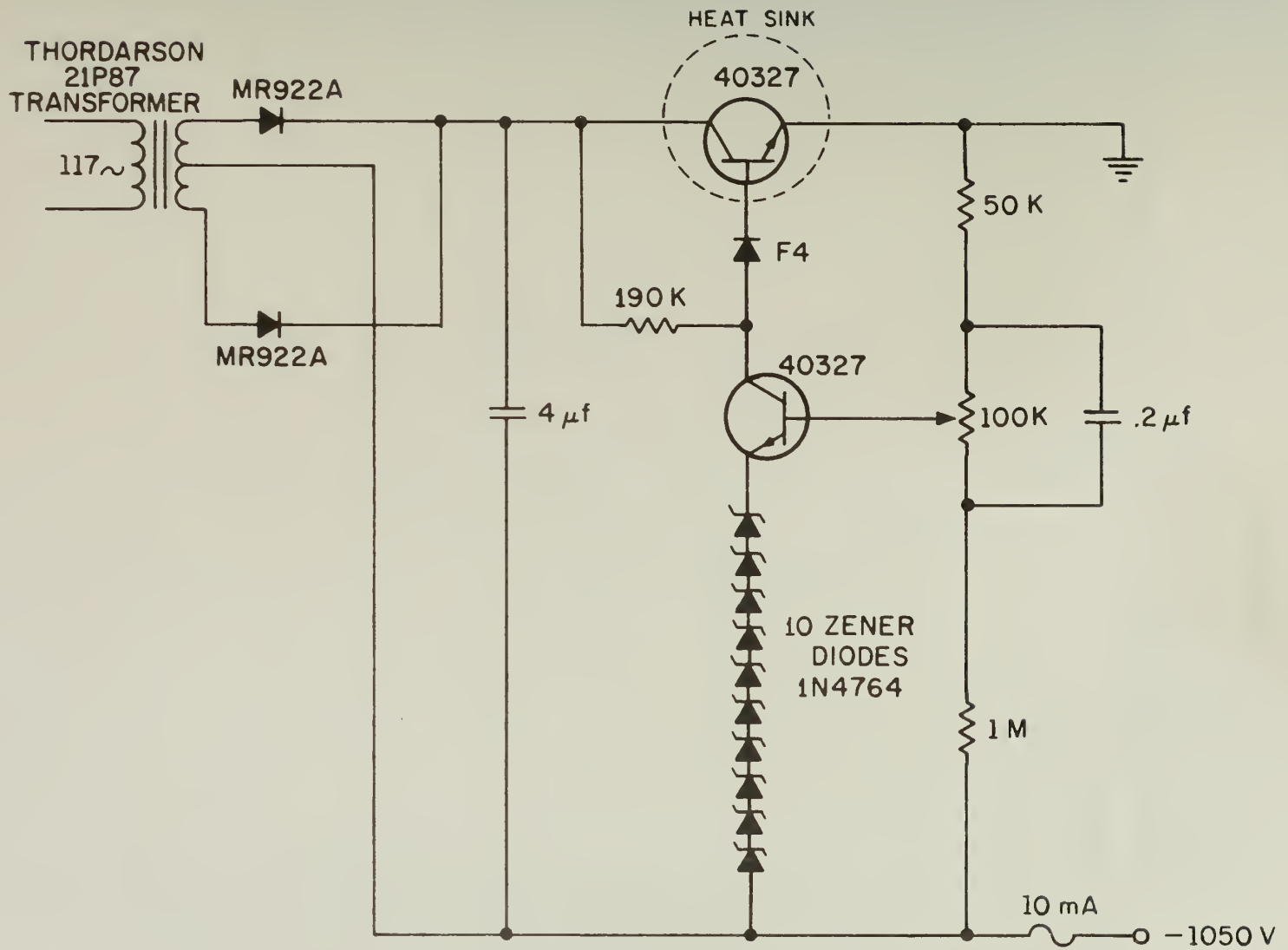

Figure 2.8 Photomultiplier Power Supply

Figure 2.9 shows the 5 volt power supply used in OCOMO. It is a straightforward use of a LM 309 5 volt output regulator. The maximum output current is 2.0 amp. A regulated 12 volt supply furnishes the input voltage. The LM 309 was used to isolate the two power supplies.

Figure 2.10 shows the -6 volt supply.

Duplicate photomultiplier, 5, and -6 volt power supplies are used in the OCOMO Decoder.

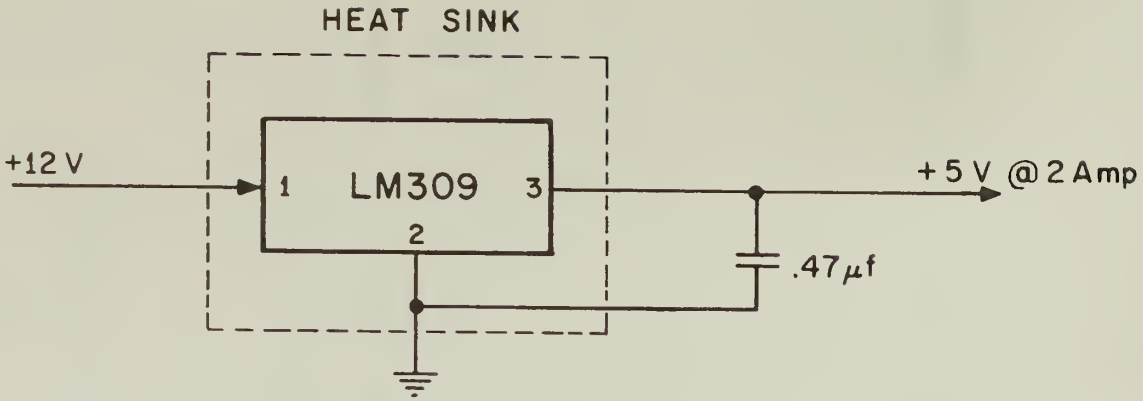

Figure 2.9 5 Volt Supply

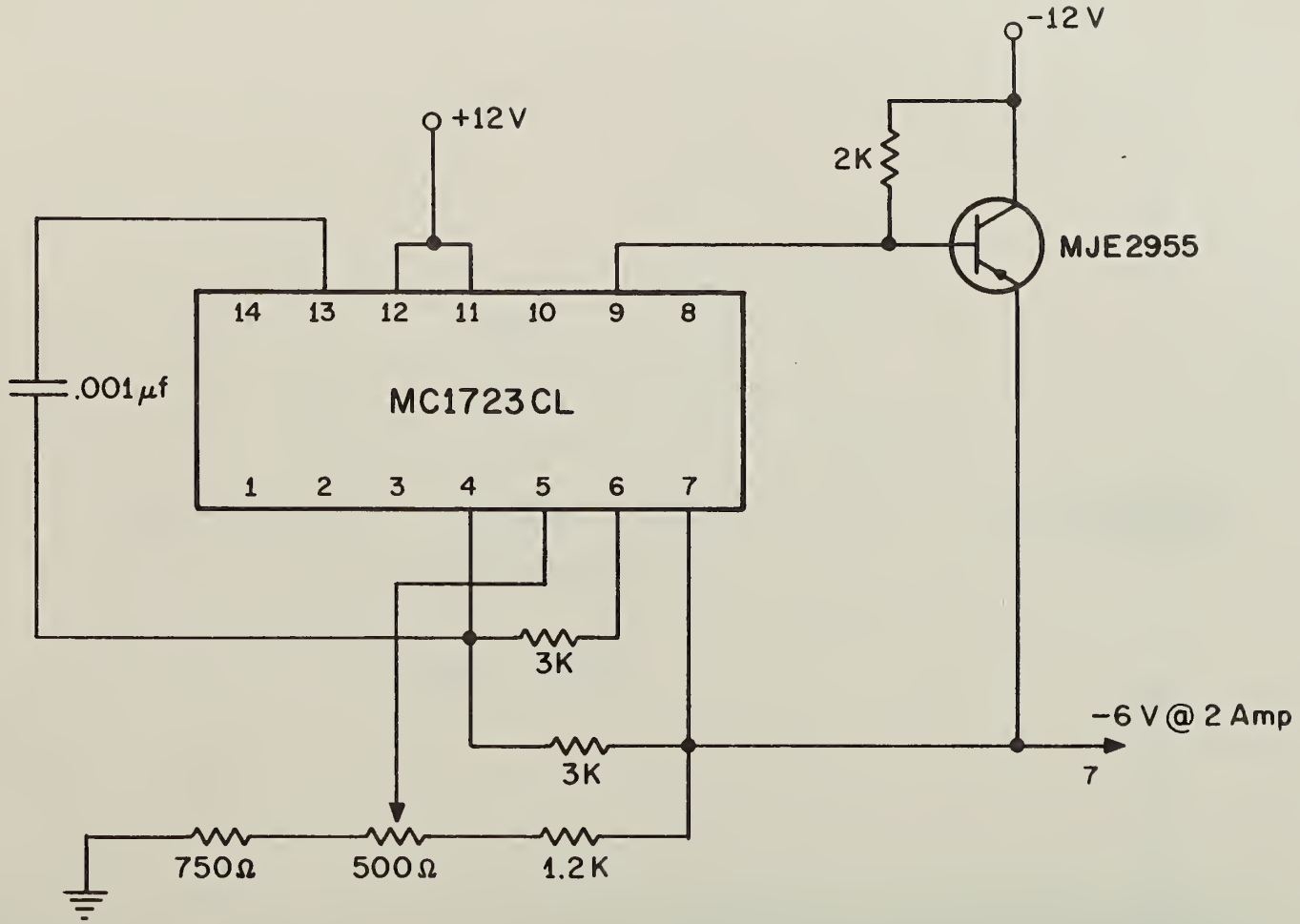

Figure 2.10 -6 Volt Supply

CHAPTER 3

OCOMO DECODER, A CIRCUIT DESCRIPTION

3.0 Introduction

The OCOMO Decoder block diagram is shown in Figure 3.1. As a word is being received, the position of the first "0" is stored. When the previous word has been decoded, this position is loaded into a register for conversion into an analog voltage. This voltage is then applied to the vertical deflection plates of the CRT. As the first row is scanned, the position of the next "0" is found. The horizontal deflection voltage is a ramp, synchronized to the vertical deflection voltage. (A ramp was used in place of a staircase because the D/A Converter required for this type of staircase circuit is very expensive)!

The decoder operates with a one codeword delay. As a new word is being received, the previous word is decoded: That is, the codeword is reassociated with its PCM level by optical correlation. The delay of 64 μs is used because of an improvement in performance over no delay in decoding. With no delay, the longest time available for scanning a row is 10 μs. Every codeword has four "0"'s and four "1"'s. The rows of the codestrips corresponding to the bit positions of the "0"'s in the codeword are scanned for decoding. If the "0"'s are in the last four positions, the scanning of the four rows could begin after the fourth "1" is received. This allows about 40 μs to scan the four rows, and therefore 156 ns for each bit in a row. The shortest persistence fall time of a readily available CRT phosphor is about 120 ns. If the no delay procedure is used, there would be an interval of 30 - 40 ns in which the data would be reliable. Such a requirement would necessitate tight specifications for other sub-systems in the decoder. When considering extra cost, lack of real benefit with a no delay procedure, and poorer performance, a one codeword delay is the obvious answer.

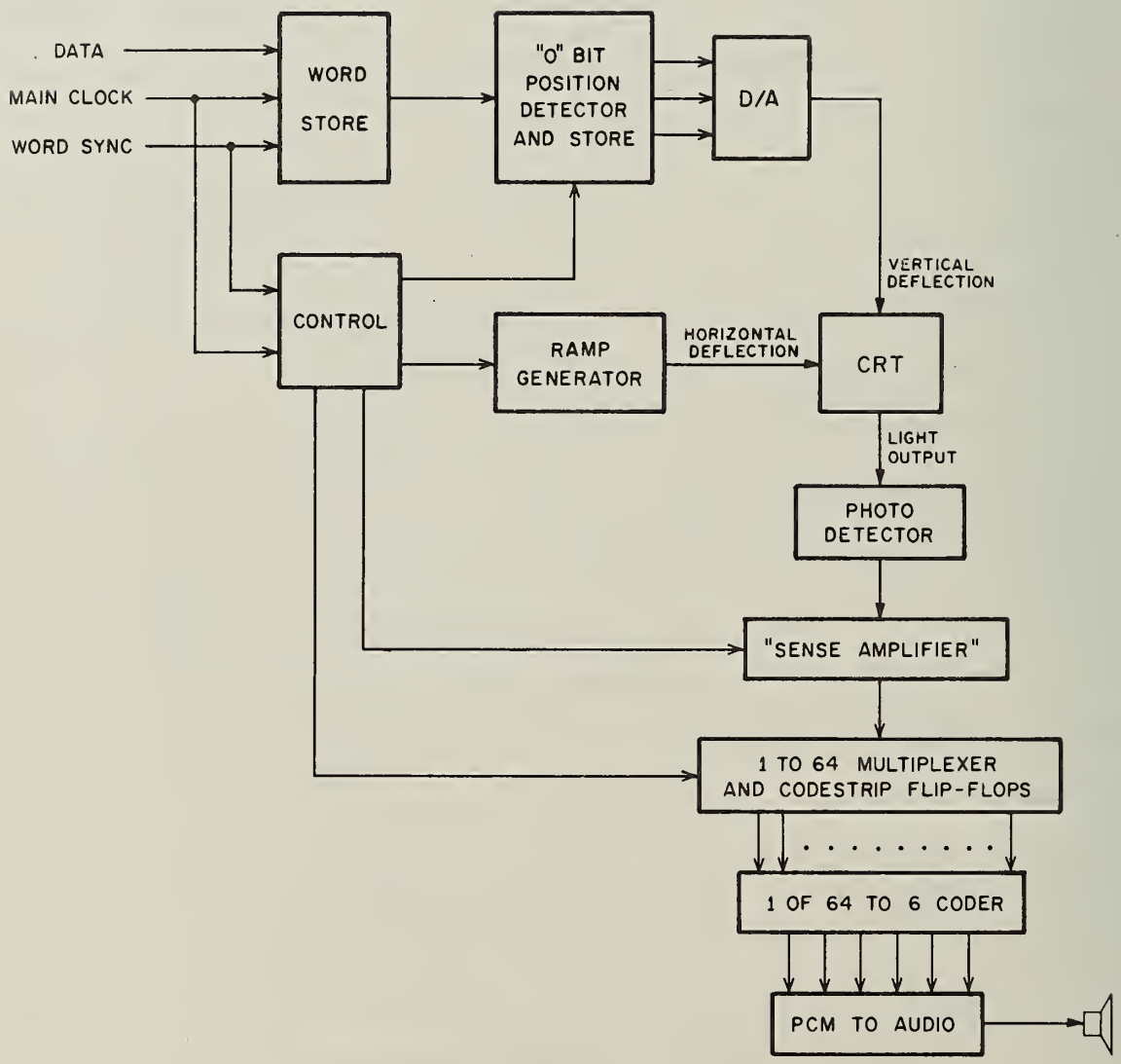

Figure 3.1 Encoder Block Diagram

The photodetector to sense the light from the CRT is a photomultiplier tube. The "sense amplifier" consists of a TTL gate and a flipflop controlled by the system clock to store the data between clock pulses. The photomultiplier acts as a current switch to turn the TTL gate on and off.

Associated with each of the 64 codestrips is a flipflop. When decoding begins, each flipflop is initialized. When the beam is behind the codestrip associated with a particular flipflop, the flipflop is enabled, and changes state if light is detected. After all four rows are scanned, only one of the 64 flipflops is in the initial state! Each flipflop is numbered between 0 and 63. After decoding, the flipflop in the initialized state has its number converted to a 6-bit representation by a 1-of-64-to-6-coder. This 6-bit number is stored and converted to an analog signal. After filtering, the signal is then amplified to drive a speaker.

3.1 Word Store Card

The Word Store Card circuit is shown in Figure 3.2. There are two modes of codeword synchronization as discussed in section 2.3. If the sync is combined with the data, two 72510 voltage comparators are used. One is always used to determine the incoming data bit state. The other comparator discriminates the sync from the data by checking for pulses 2 volts above the normal "1" data level. The other mode uses a sync pulse on a separate line: When a sync pulse is received by either method, the Word Counter is loaded with "1101". The choice of the 2^3 bit as "1" is arbitrary, since it assigns the count-up mode of the encoder staircase generator to this value. The three lower order bits are determined by the sync pulse occurring during the binary "100" bit position of the codeword. The binary number "101" is loaded into the lower three bits of the counter since the counter is not loaded until the clock pulse occurring during the binary "101" bit position of the codeword.

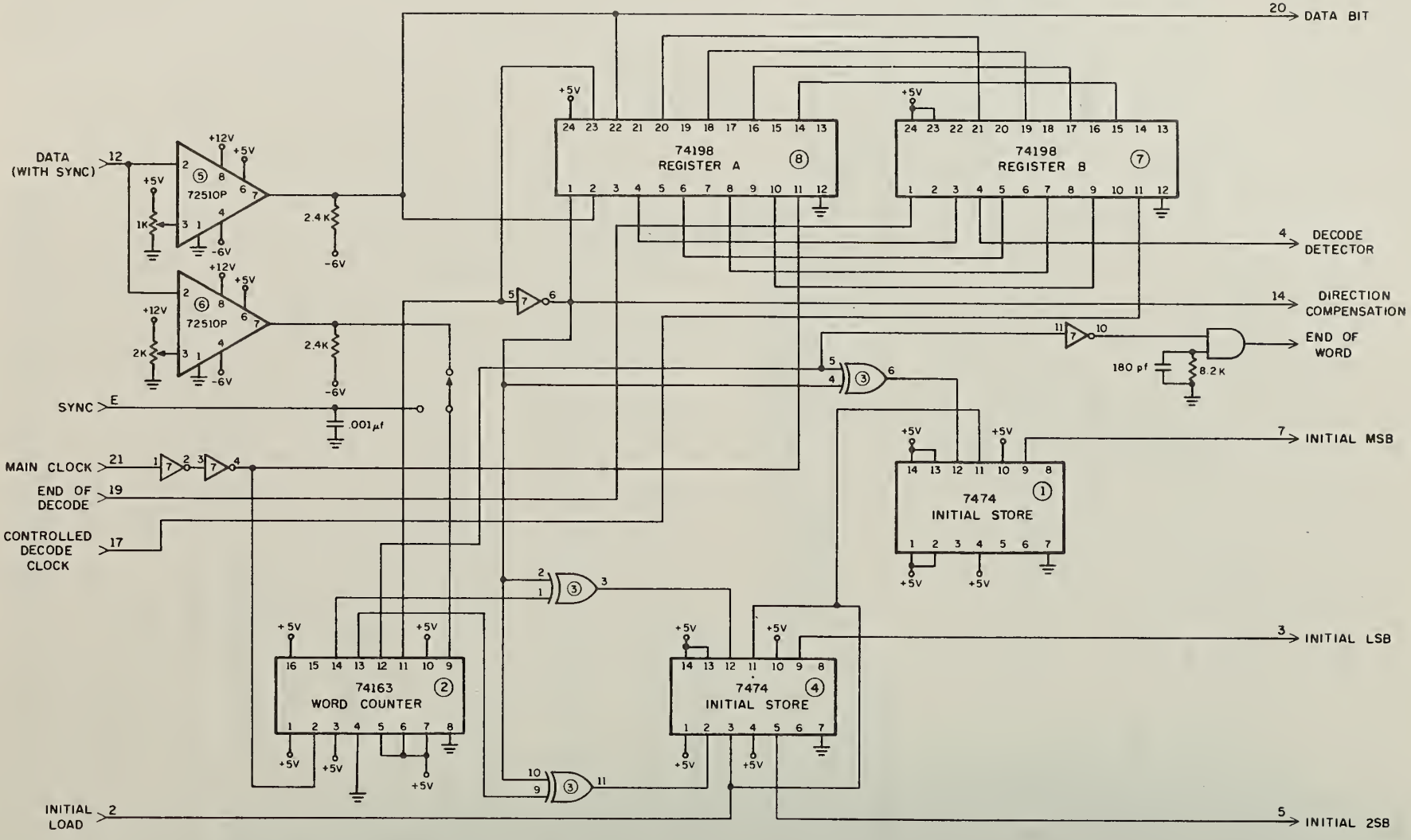

Figure 3.2 Word Store Card

Register A is used as a serial-to-parallel-converter. Since the order of the bits changes direction every codeword, the direction of Register A's shifting is changed. The highest order bit of the Word Counter determines the direction of shifting. After the last bit is received, the new word is loaded into Register B.

As the new codeword is being received in Register A, the lowest valued "1" position is stored in the Initial Store Flipflops. (The "1" position is actually a "0" position since the inverse of the received word is shifted in Register A!) Since the direction of the bits changes every other codeword, some complications arise in storing the lowest valued "1" position. If the bits occur in increasing order, the first "1" position must be stored, and the Word Counter then contains the value of the bit position. If the bits occur in decreasing order, the last "1" position must be stored, and the Word Counter now contains the bit-by-bit inverse of the position. When the bits are increasing, only the first "1" generates an Initial Load pulse. Otherwise each of four "1" positions generates an Initial Load pulse. These pulses are generated on the Vertical Voltage Card described in the next section. The inversion of the bits is done with two-input EXCLUSIVE-OR gates. If the bits are increasing, one input of each EXCLUSIVE-OR gate is at "0". The other input comes from the Word Counter. Thus, the bits are not inverted. If the bits are decreasing, the common input is one, and the bits will be inverted.

After the new codeword is loaded in Register B and the lowest valued "1" position is loaded into a register on the Vertical Voltage Card, the next "1" position must be found. This is accomplished by shifting the B Register. The first "1" position is ignored and the register is shifted until the next "1" appears at the Decode Detector, the signal which controls the shifting circuitry on the Vertical Voltage Card. After the second position is

transferred, the next "1" position is found by shifting. Similarly, the fourth "1" is found.

3.2 The Vertical Voltage Card

The circuit for the Vertical Voltage Card is shown in Figure 3.3. The Initial Load signal, which loads the lowest order "1" position on the Word Store Card, is generated by the AND of the Data Bit signal and the delayed positive edge of the Main Clock from the encoder. The AND of the two signals is used to ensure that the Word Counter on the Word Store Card has settled before bits are stored. If the Direction Compensation signal is low, all four of the "1" bits of the incoming word produce Initial Load pulses. If the Direction Compensation signal is a "1", only the first "1" produces a pulse. This Initial Load pulse changes the state of a flipflop which then inhibits the signal until the next decoding cycle.

After decoding a word, the Vertical Counter is loaded with the lowest valued "1" position of the word just received. Shortly after this, the End of Row Decode signal causes the number stored in the counter to be loaded into the Vertical Store Flipflops. The outputs of the flipflops are converted to an analog signal and applied to the vertical deflection amplifier of the CRT. This determines the first row of the codestrips to be scanned. While the row is scanned, the Controlled Decode Clock shifts Register B of the Word Store Card. When the first "1" occurs at the Decode Detector, the clock to the Vertical Counter is enabled as one of the 7473 flipflops changes state. Register B is shifted until the next "1" occurs. The Vertical Counter then contains the position of this "1", and the Controlled Decode Clock is inhibited. At the end of the scanning, the contents of the Vertical Counter are transferred to the Vertical Store Flipflops for D/A conversion. The Controlled Decode Clock is enabled by the falling edge of the End of Row Decode signal.

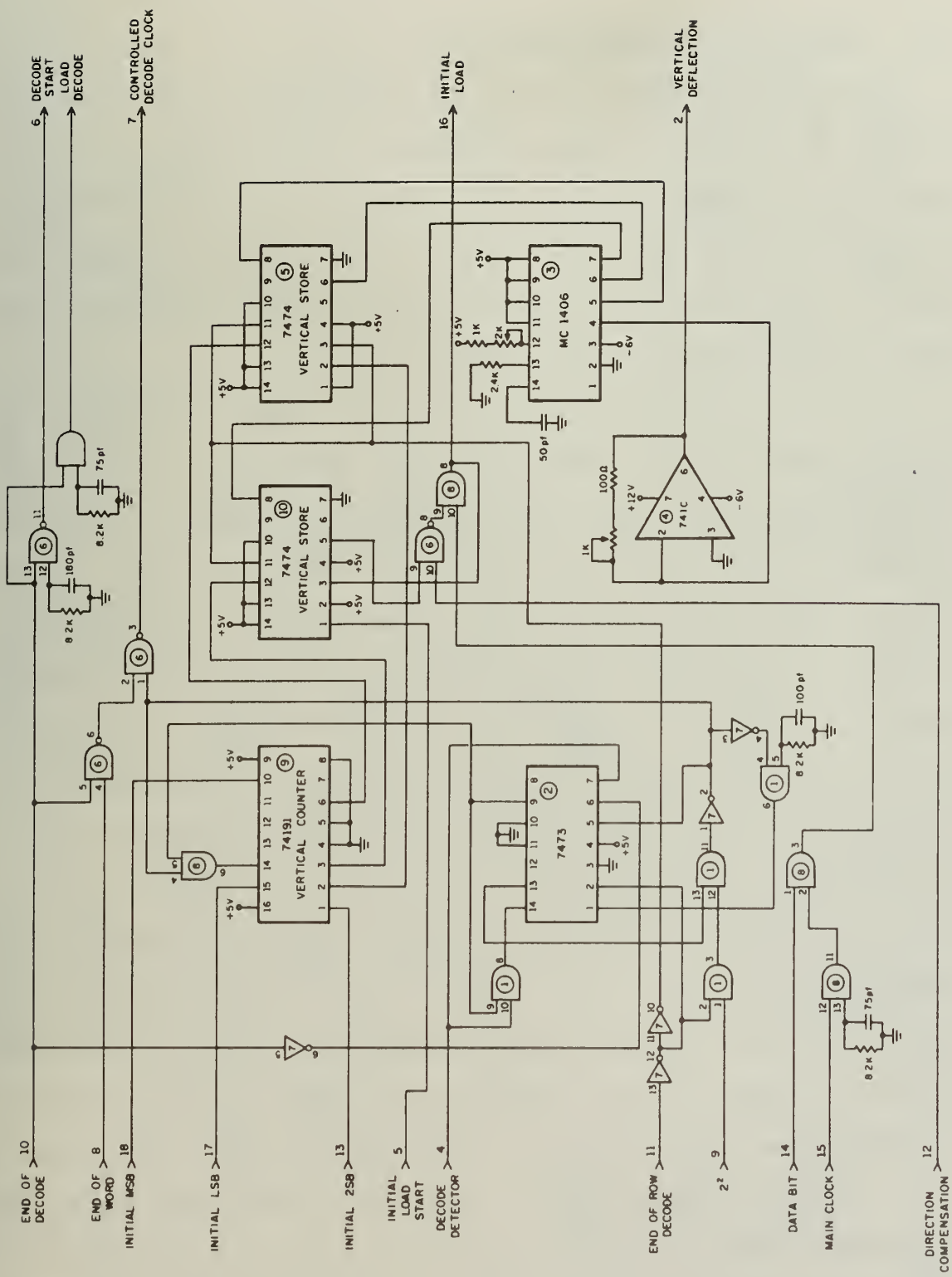

Figure 3.3 Vertical Voltage Card

The clock continues until the third "1" is found, and the same process occurs as for the second "1". The same procedure is used for the fourth "1".

3.3 Horizontal Ramp and Control Card

The circuit diagram for this card is shown in Figure 3.4, and key waveforms are depicted in Figures 3.5 and 3.6. Figure 3.5 shows the waveforms during the length of 2 codewords. Figure 3.6 shows the waveforms during one Horizontal Ramp. In Figure 3.6 there are waveforms labeled clock D and D': The Clock D waveform applies during the first 3 Horizontal Ramps; Clock D' applies during the fourth Horizontal Ramp.

The falling edge of the End of Word signal marks the end of the codeword just received and the beginning of a new codeword. The rising edge of the Decode Start signal causes the discharging of the Horizontal Ramp generating capacitor. The rising edge of the Delayed Decode Start signal initializes the circuit and starts the Decode Clock. The Horizontal Counters count from 0 to 63. When the counter reaches 63, the MAX/MIN signals of the counters become "1"'s since the two highest order bits are initially loaded as "1"'s. The Q output of the Delay F.F. becomes a "1" after the falling edge of the next Decode Clock pulse. This inhibits the counters and generates an End of Row Decode pulse. This End of Row Decode pulse stops the charging of the ramp generating capacitor. After a one monostable triggering delay, the discharging of the capacitor begins. After a delay of about 5 clock pulses, the positive edge of the Q flipflop appears at the output of the AND Gate Delay and loads the counter to the "11000000" state. This causes the MAX/MIN signal of the higher order bit counter to become a "0". On the next falling edge of the clock, the Q of the Delay F.F. becomes a "0" and enables the Horizontal Counters. Also, the negative edge of Q increments a two bit counter. The delay of five clock pulses, about 1.25 μs, is needed to ensure the vertical

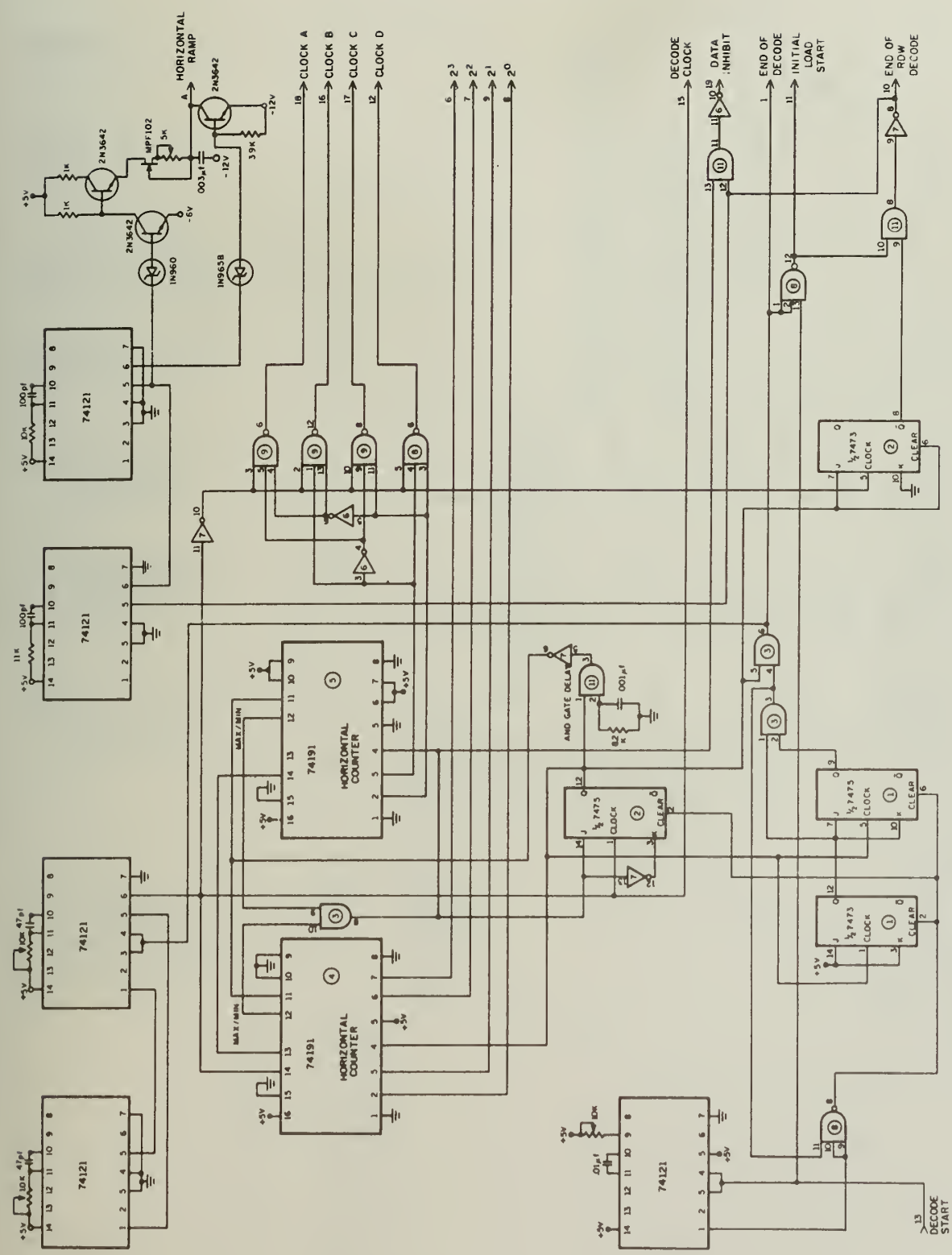

Figure 3.4 Horizontal Ramp and Control Card

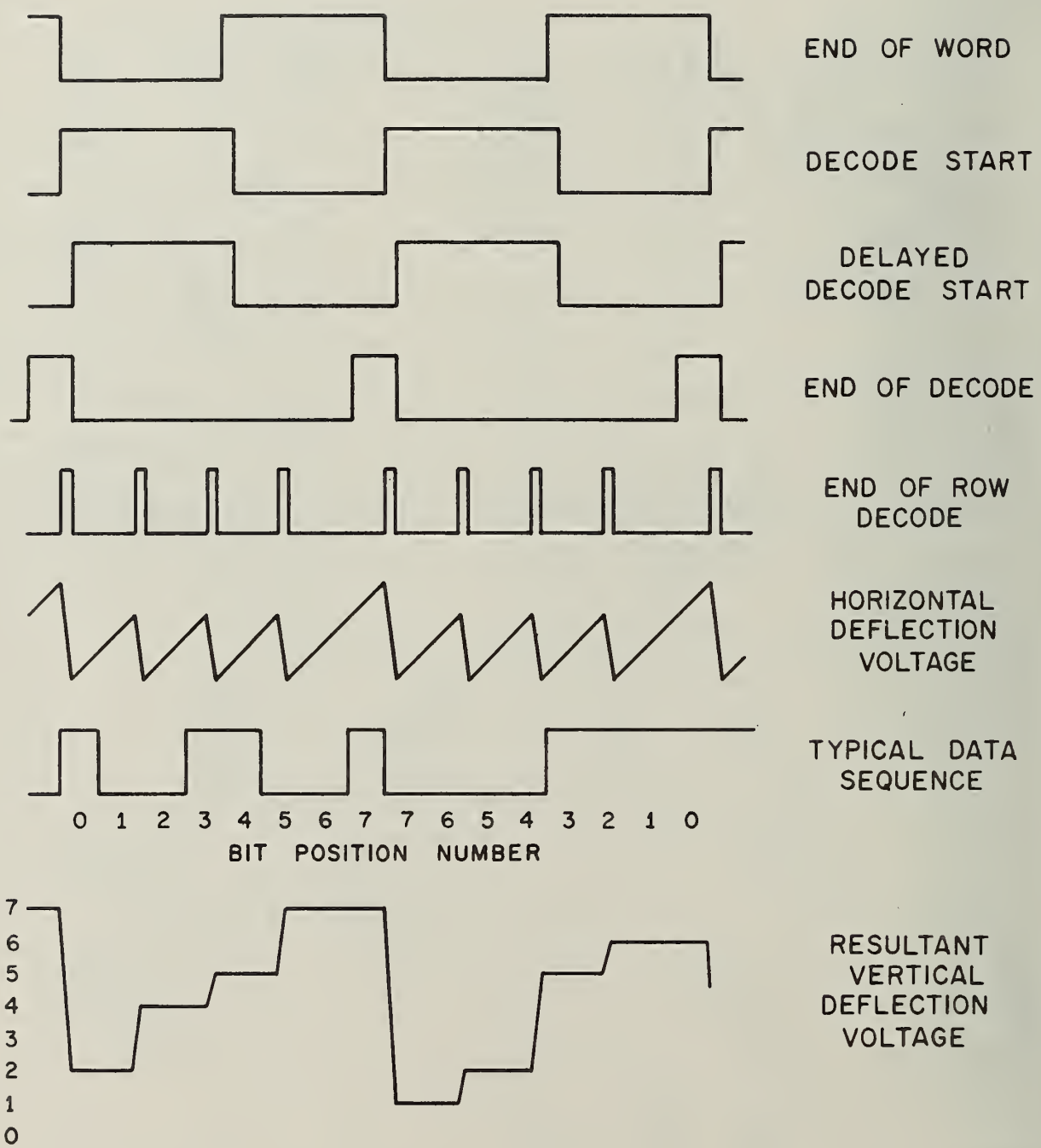

Figure 3.5 Decoder Waveforms

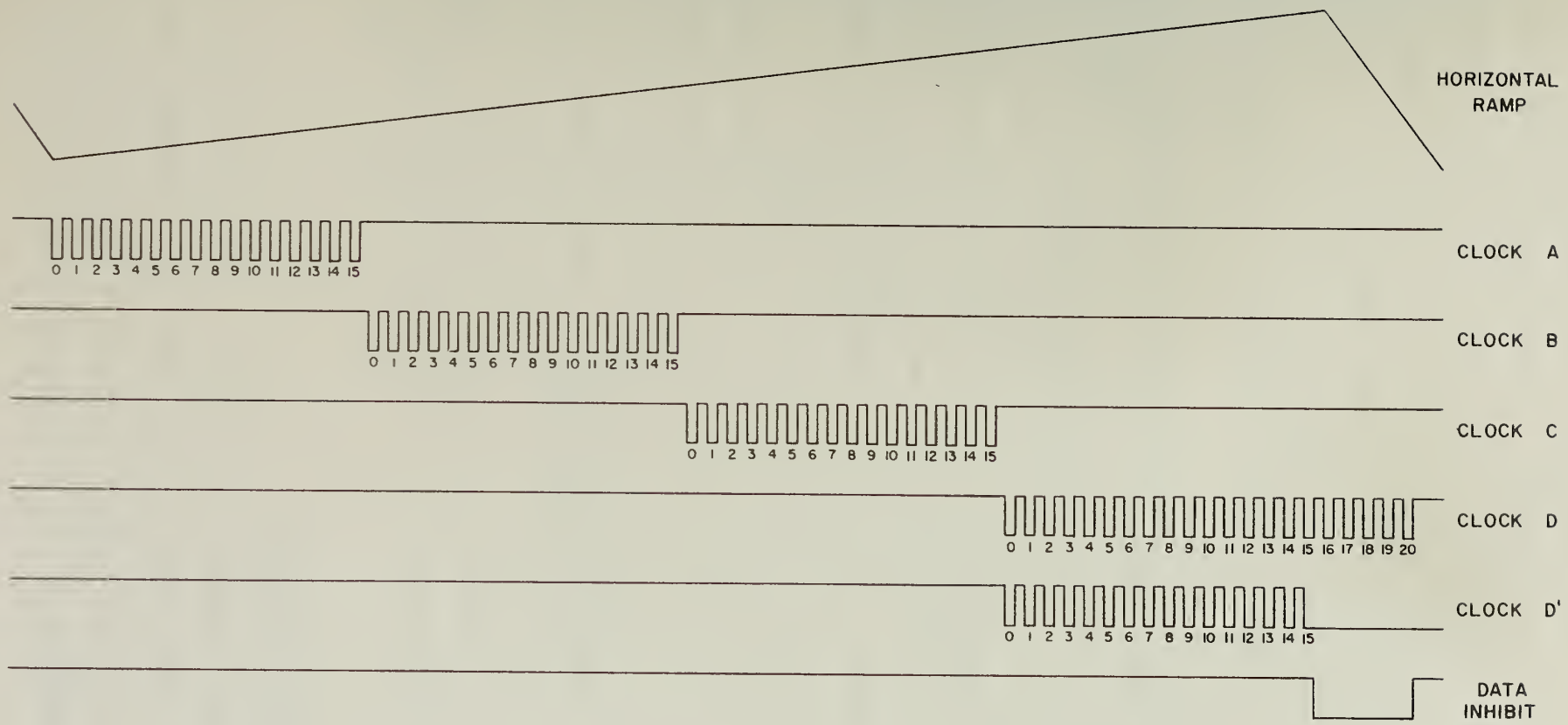

Figure 3.6 Decoding Waveforms

D/A and related circuitry has settled. The same cycle occurs 3 more times during the decoding of the codeword. At the end of the fourth cycle, the two bit counter is in the "11" state. When Q of the Delay F.F. becomes a "1" at the end of the fourth cycle, the End of Decode signal becomes a "1" and inhibits the Decode Clock. The Horizontal Ramp continues to rise until the leading edge of the Decode Start signal occurs; this causes the End of Row Decode signal to become a "1". The leading edge of the Delayed Decode signal starts the first of four cycles.

The Horizontal Ramp generator consists of a MPF102 FET, used as a constant current source to charge a capacitor. The horizontal deflection voltage is generated by controlling the current to the current source. In charging, the current from the source is integrated by the capacitor. In discharging, the current to the source is turned off, and a transistor discharges the capacitor. The output is taken directly from the capacitor, since it goes into a high input impedance oscilloscope amplifier.

The last two waveforms of Figure 3.5 depict a typical data sequence and the resultant vertical deflection voltage. The vertical voltage during the first codeword is the result from the previous codeword (which is not shown in this figure). The output during the second codeword is the result from the first codeword shown. Since there are "0"'s in the first, second, fifth and sixth bit positions, the vertical deflection voltage waveform deflects the beam behind the first, second, fifth and sixth rows of the code-strip mask.

3.4 The "Sense Amplifier"

The "sense amplifier" circuit converts the current pulses of the photomultiplier tube into a digital voltage. The digital voltage is then stored during the center of the current pulses with a flipflop. The circuit diagram is shown in Figure 3.7.

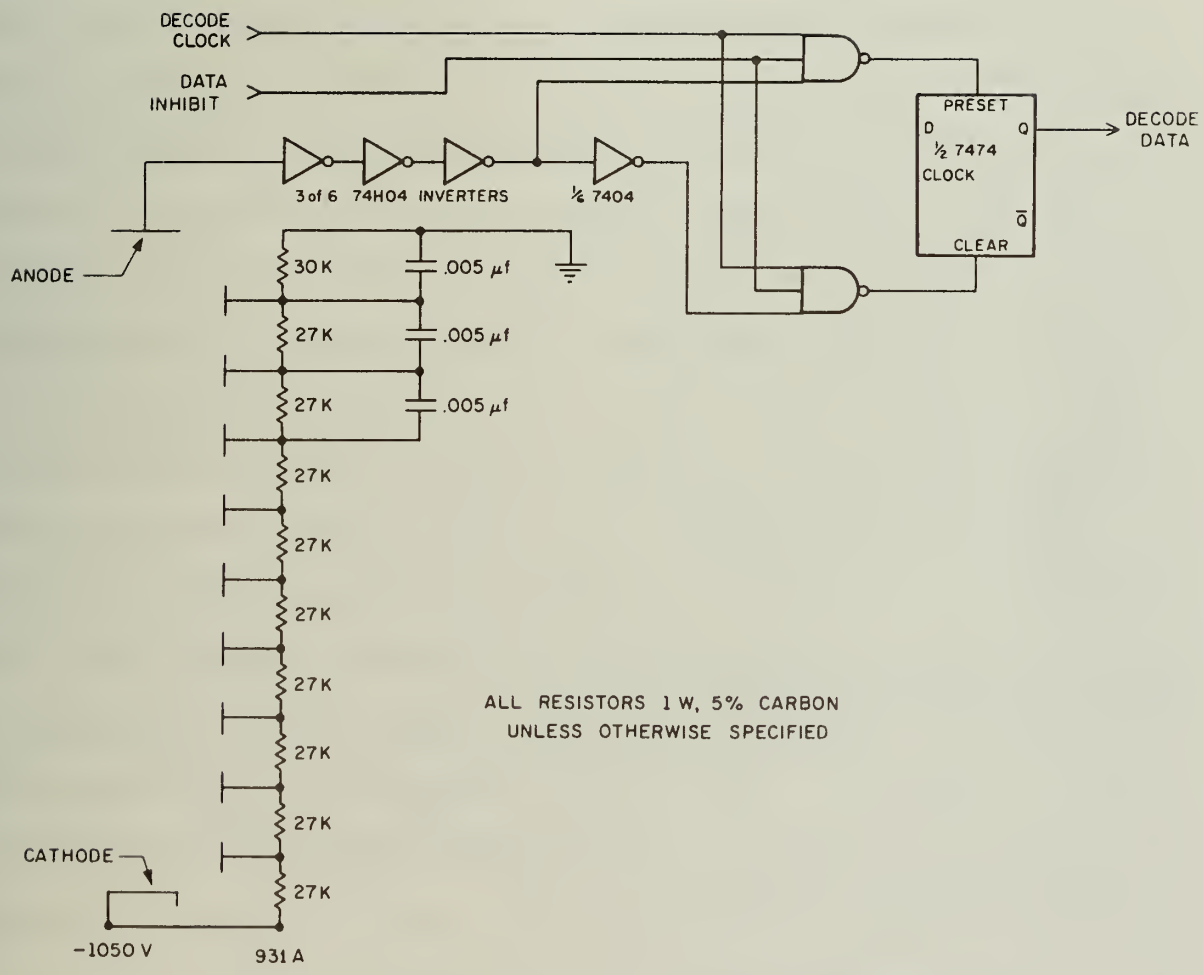

Figure 3.7 Encoder Photomultiplier and "Sense Amplifier"

The photomultiplier tube is an RCA 931A. The maximum average anode current of this tube is 1 ma. The current risetime of the anode current is 2 ns. A high speed TTL gate input puts out a current of 2 ma. Since half of the codestrip positions pass light, the average current the photomultiplier would sink from a high speed TTL gate is 1 ma. Also, since the TTL gate has good noise immunity and speed characteristics, it makes an ideal "interface" between the photomultiplier tube and TTL logic.

The amplitude and DC level of the Horizontal Ramp is adjusted so that the CRT beam is in the center of an opaque or transparent rectangle of a codestrip when the Decode Clock makes a positive transition: This adjustment makes the data most reliable, since the effects of drift or non-linearity are minimized. The data from the "sense amplifier" is stored at the most "reliable time," so as to optimize performance. The data is stored by activating the preset or clear of the 7474 flipflop. The flipflop can change state only if the Data Inhibit line is in the inactive state ("1") and the Decode Clock is "1". If the above two conditions are met, and no light activates the photomultiplier tube, the preset is activated. If light is present, the clear is activated. The Data Inhibit signal is used to stop the flipflop from changing during the first three falling edges of the Horizontal Ramp occurring in decoding a codeword. The flipflop does not change state on the falling edge of the ramp prior to the start of decoding because the Decode Clock signal is a "0" after decoding is completed.

3.5 The Decoding Circuitry

Decoding is achieved by scanning the rows of the codestrips as determined by a received codeword. This scanning of the rows reassociates the received word with its PCM level: The codestrip associated with the received word will pass no light. Associated with each codestrip is a flipflop. After the associated codestrip flipflop is found, the proper PCM level is generated

and a PCM to analog conversion completes the decoding. Figure 3.8 shows one of four identical circuits which contain 16 codestrip flipflops and a 1-of-16-to-4 coder.

The F.F. Clear signal, a 200 ns pulse triggered by the rising edge of the Decode Start signal, clears all the codestrip flipflops just prior to the start of decoding. The 74154 4-to-16 multiplexer is controlled by the Decode Clock. One of 16 lines is selected by the four bits of the lower order Horizontal Counter of the Horizontal Voltage card. One of the four circuits is selected by Clock A, B, C, or D. In order for a Q of a particular codestrip flipflop to make a positive transition, the particular Clock A, B, C, or D must be a "0", the Decode Data signal must be a "0", and the 1 of 16 lines associated with this particular flipflop must be selected. These conditions correspond to light passing through the codestrip associated with the flipflop. The flipflop is connected in the following manner: The lines of the multiplexer are connected to the clock inputs of the 7473 dual JK flipflops. The J's are at "1" and the K's are at "0". If the multiplexer line makes a negative transition, the Q of that flipflop will be a "1".

After decoding, one of the codestrip flipflops will have its Q in the "0" state because its clock never received a falling edge. The 1-or-64-to-6 encoding then begins on the Codestrip Flipflop Card. On each of the cards is a 1-of-16-to-4 coder; a minimal package court is achieved by using 8-input-NAND gates for the coder. The Q's of the codestrip flipflops are connected to the inputs of the appropriate NAND gates, in order to generate the bit-by-bit inverse of the binary number which selects the line associated with the particular Q. For example, if the Q associated with the "0000" line is a "0", the outputs of the 4 NAND gates will be "1111".

The final encoding is completed by the 20-to-6 Coder circuit of Figure 3.9. In this circuit the 4 NAND gate outputs of each of the four

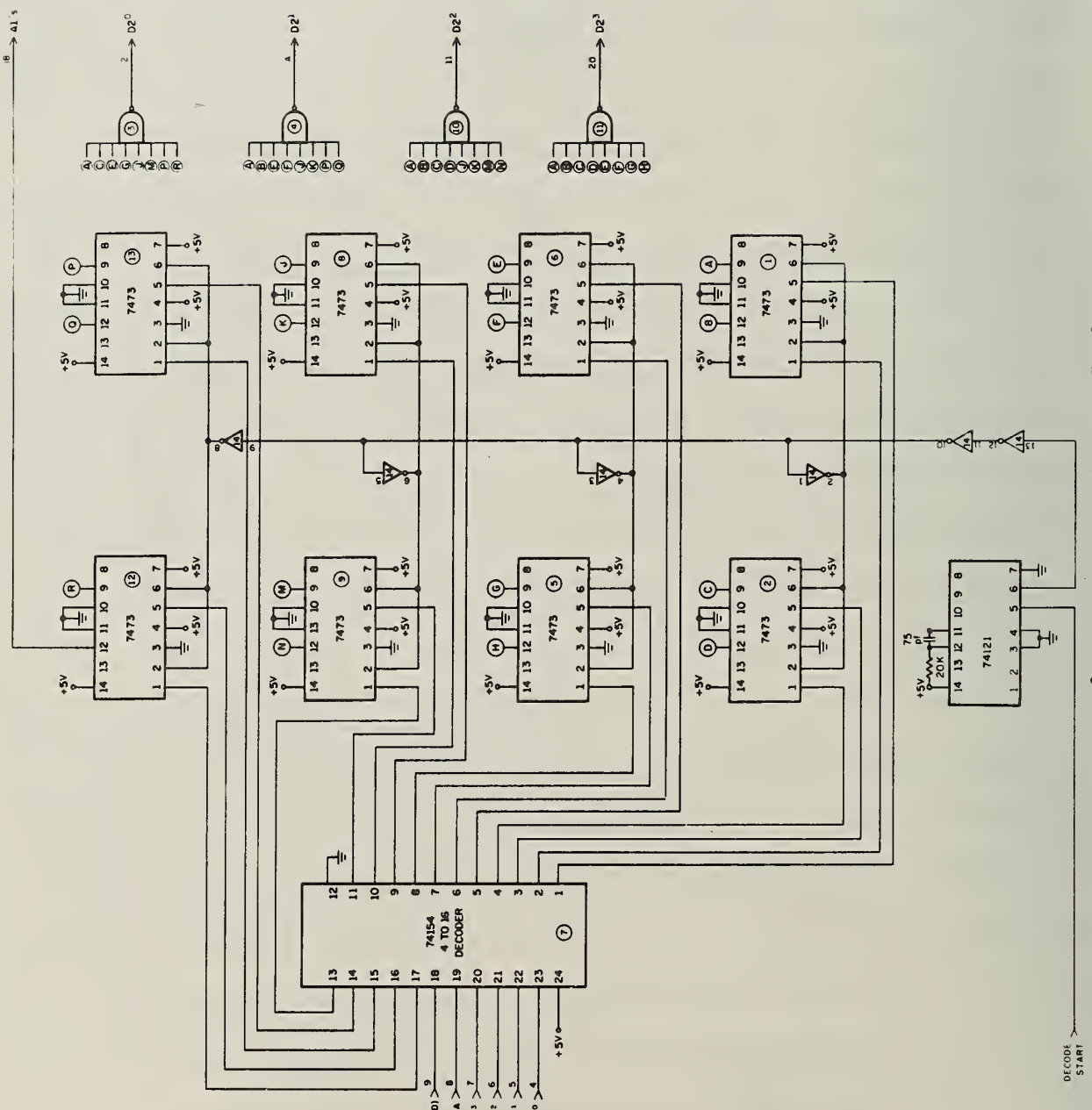

Figure 3.8 Codestrip Flipflop Card

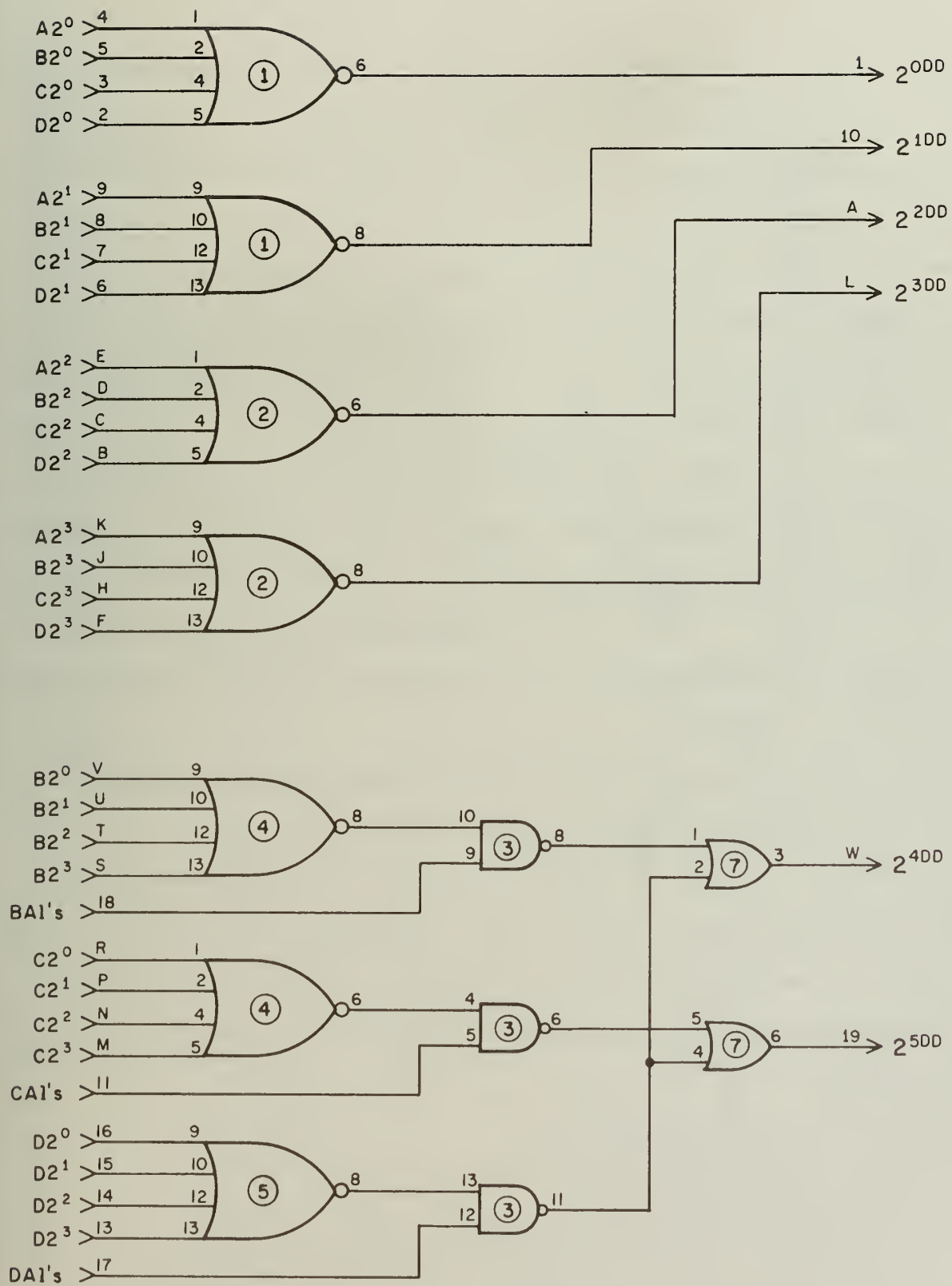

Figure 3.9 20-to-6 Coder

identical circuits are combined to determine the four lower order bits. The upper two bits are determined by which one of the four cards contains a Q in the "0" state. A 4-input NOR gate is used to combine the four lower order bits. The letter prefixes of the inputs signify which of the four circuit cards the input comes from. Since only the inputs from the card with a Q in the "0" state can be a "1", the inputs from this circuit card determine the four lower order bits. The NOR gates again perform the bit-by-bit inversion of the NAND outputs. The same idea is used to determine the upper two bits, but there is an added complication: Since the "0000" NAND gate output corresponding to the "1111" line would be undetected, the (B, C or D) 1's signal must be included. This coding of the two higher order bits is accomplished by NORing the NAND outputs from one circuit card. The output of the NOR is NANDed with the (B, C or D) 1's signal. The output of the NAND will be a "1" only if there is a Q in the initialized state on the related circuit card. The output of the NAND gates are then connected to OR gates in order to generate the proper output.

Now the received codeword has been reassociated with its PCM level. This PCM signal is then converted to an analog signal using the circuit of Figure 3.10. The 6-bit PCM signal is loaded by the delayed rising edge of the End of Decode signal into the 74198 shift-register. This signal is then converted to an analog signal by an MC 1406 D/A Converter, and the analog signal is then filtered by a low-pass active filter. An audio amplifier drives the speaker.

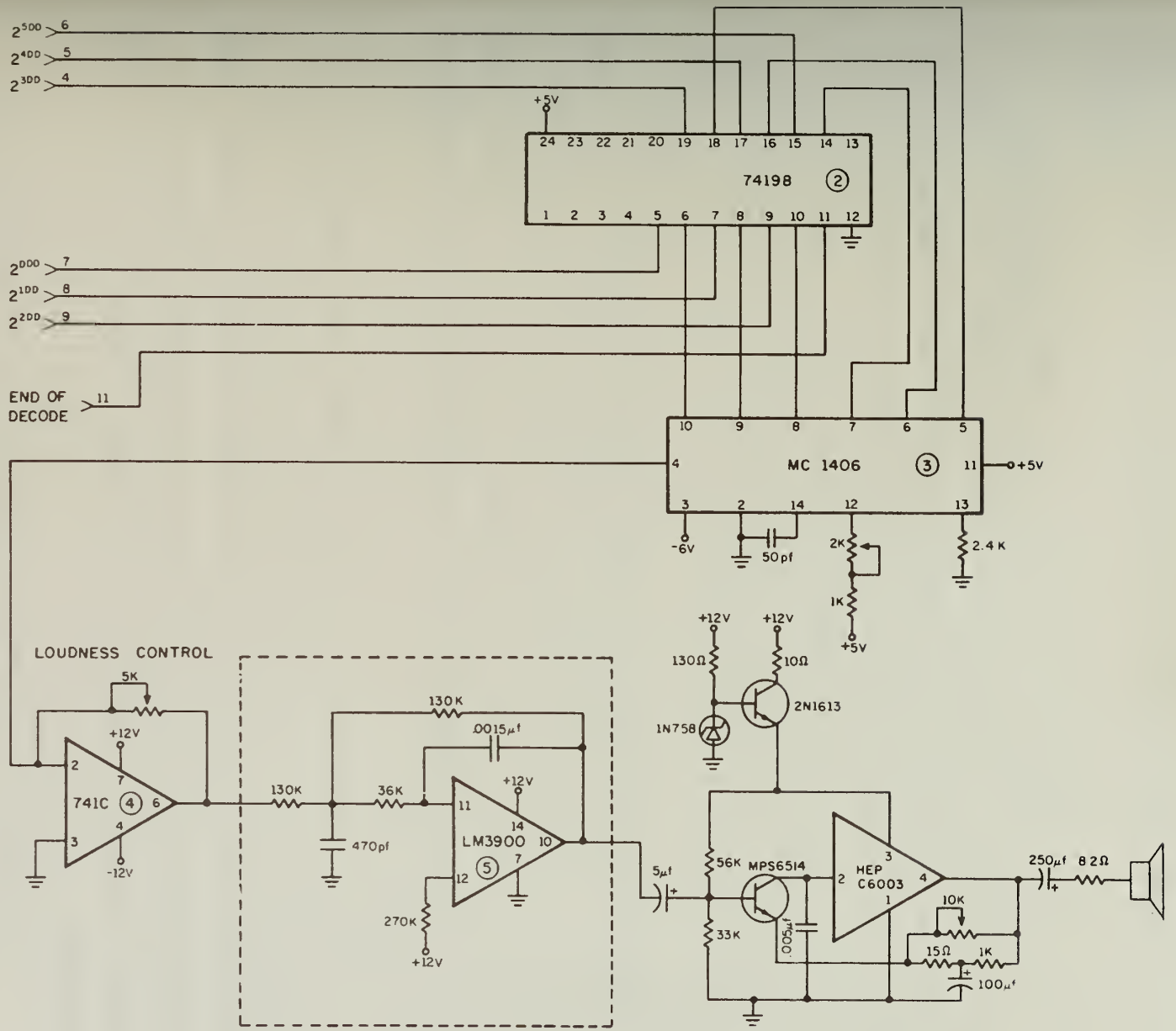

Figure 3.10 PCM to Voice Converter

CONCLUSIONS

There are 64! permutions or about 1.27×10^{89} different arrangements of a set of 64 codewords. Since there are $\binom{8}{4} = 70$ different possible codewords, there are $\binom{70}{64}$ or about 7.6×10^{8} different sets of codewords. This gives:

1.27×10^{89} code/set $\times$ 7.6×10^{8} sets or about $\underline{10^{98}}$ $\underline{\text{possible codes}}$!

Since the order of the transmission of data bits reverses after each codeword at the encoder, one of two different codes is used for asymmetric codewords. Here "symmetric" means that the first four bits are the mirror image of the last four. One of two different codes is used for asymmetric codewords because the same PCM level appears to have one of two codewords depending on the direction of the transmitted data bits: There are $\binom{4}{2} = 6$ symmetric codewords in the set of 70, therefore, any one set containing 64 codewords would have between 91% to 100% asymmetric codewords. If one makes the PCM-level-to-codeword correspondence a function of time, as described in section 1.3 (which is equivalent to choosing 1 of 1.27×10^{89} different codes for each codeword transmitted) it can be seen that deciphering the OCOMO code is a formidable task!

The optical ROM's at the decoder and encoder were designed to be basically the same. If one assembles the necessary circuitry for the encoder and decoder, one could use the same ROM for coding and decoding. This could be done by switching-in the vertical and horizontal waveforms of the encoding circuitry to code on-line. To decode, we would switch-in the decode waveforms.

The bit rate of the decode optical ROM is near its maximum, about four million bits/sec. If one wanted to increase the bandwidth of the input

signal, this optical memory could be used provided a different decoding scheme is employed. For example, one could decode by using a look-up scheme and still maintain the advantages of the optical ROM. This would increase the sample rate and therefore the bandwidth by a factor of 43 for the present bit rate, since six versus 256 bits would be read out of the memory for each codeword.

The main advantage of the optical ROM used in OCOMO is that the information in the memory can be changed simply, compared to an electrical ROM. One way to change the contents of the ROM would be to have the codestrips stored on spools much like many cameras store film. One could then arrange the codestrips, so the ROM information could be changed by turning the spools in synchronism at the encoder and decoder. The strips could be placed, so that the code mask could be moved one or more codestrips across the face of the CRT...

In conclusion, the optical ROM used in OCOMO provides flexibility and speed not available in other types of memories.

LIST OF REFERENCES

1. Carlson, A. B., Communication Systems: An Introduction to Signals and Noise in Electrical Communication. McGraw Hill Book Company, 1968.

2. Embinder, J., Fet Applications Handbook. Tab Books, 1970.

3. Schwartz, M., Information Transmission, Modulation, and Noise. McGraw Hill Book Company, 2nd ed., 1970.

4. RCA Electron Tube Handbook, Radio Corporation of America. Harrison, New Jersey.

5. Photomultiplier Manual, Radio Corporation of America, 1970.

6. The Integrated Circuits Catalog, Texas Instruments, Incorporated.

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)*
Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, Illinois 61801

2a. REPORT SECURITY CLASSIFICATION
Unclassified

2b. GROUP

3. REPORT TITLE

OCOMO: OPTICAL CORRELATION MODULATION

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*
Master's Thesis, Technical Report March, 1974

5. AUTHOR(S) *(Last name, first name, initial)*

Budzinski, Robert L.

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
March, 1974	54	6

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
N000 14-67-A-0305-0007 b. PROJECT NO.	
c.	9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*
d.	

10. AVAILABILITY/LIMITATION NOTICES

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Office of Naval Research 219 South Dearborn Street Chicago, Illinois 60604

13. ABSTRACT

An optical read only memory is used to digitally encode a pulse code modulated voice signal. Decoding is accomplished by using an optical read only memory to correlate the received codeword with the set of codewords.

DD FORM 1 JAN 64 1473

UNCLASSIFIED
Security Classification

BIBLIOGRAPHIC DATA SHEET	1. Report No. UIUCDCS-R-74-631	2.	3. Recipient's Accession No.
4. Title and Subtitle OCOMO: OPTICAL CORRELATION MODULATION			5. Report Date March, 1974 6.
7. Author(s) Robert L. Budzinski			8. Performing Organization Rept. No. UIUCDCS-R-74-631
9. Performing Organization Name and Address Department of Computer Science University of Illinois at Urbana-Champaign Urbana, Illinois 61801			10. Project/Task/Work Unit No. 11. Contract/Grant No. N000 14-67-A-0305-007
12. Sponsoring Organization Name and Address Office of Naval Research 219 South Dearborn Street Chicago, Illinois 60604			13. Type of Report & Period Covered Technical 14.

15. Supplementary Notes

16. Abstracts

An optical read only memory is used to digitally encode a pulse code modulated voice signal. Decoding is accomplished by using an optical read only memory to correlate the received codeword with the set of codewords.

17. Key Words and Document Analysis. 17a. Descriptors

Private communication

Pulse code modulation

Optical read only memory

17b. Identifiers/Open-Ended Terms

17c. COSATI Field/Group

18. Availability Statement	19. Security Class (This Report)	21. No. of Pages
Unlimited distribution	UNCLASSIFIED	
	20. Security Class (This Page) UNCLASSIFIED	22. Price

FORM NTIS-35 (10-70) USCOMM-DC 40329-P71